

Ícaro Meneses Ferreira de Santana

**Projeto de um conversor analógico-digital de  
registrador de aproximações sucessivas em  
tecnologia de 180nm**

Campina Grande, Paraíba

Dezembro de 2024

Ícaro Meneses Ferreira de Santana

## **Projeto de um conversor analógico-digital de registrador de aproximações sucessivas em tecnologia de 180nm**

Relatório de Estágio Integrado submetido à Coordenadoria de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica

Universidade Federal de Campina Grande – UFCG

Centro de Engenharia Elétrica e Informática – CEEI

Departamento de Engenharia Elétrica – DEE

Orientador: Georgina Karla Freitas Serres, Dra.

Campina Grande, Paraíba

Dezembro de 2024

Ícaro Meneses Ferreira de Santana

## **Projeto de um conversor analógico-digital de registrador de aproximações sucessivas em tecnologia de 180nm**

Relatório de Estágio Integrado submetido à Coordenadoria de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica

Trabalho aprovado. Campina Grande, Paraíba, Dezembro de 2024:

---

**Georgina Karla Freitas Serres, Dra.**  
Orientadora

---

**Marcos Ricardo Alcantara Morais, Dr.**  
Convidado

Campina Grande, Paraíba  
Dezembro de 2024

*Aos meus pais, Maria do Socorro e Vanilson, que me proporcionaram alcançar novos horizontes.*

# Agradecimentos

Ao concluir esta fase decisiva de minha formação acadêmica, expresso minha gratidão a todos que de maneiras distintas contribuíram para o meu crescimento, oferecendo apoio ora como encorajamento, ora como desafio. Cada contribuição, não importa a forma, foi essencial para que eu pudesse chegar até aqui.

Aos meus pais, verdadeiros alicerces de minha existência, agradeço pela educação e pelo amor inabalável. Vocês me ensinaram a enfrentar o mundo com coragem e integridade, acreditando em mim mais do que eu mesmo em diversos momentos. Obrigado por serem os primeiros e mais fervorosos apoiadores de minhas aspirações, mesmo quando estas pareciam distantes.

Aos meus amigos — Mateus Marques, Luiz Henrique, Joseilton, José Ewerton, Davi, Túlio, Lucas Farias, Hortência e aos demais do laboratório, obrigado pela companhia e pelas incontáveis horas de descontração e suporte. Cada um de vocês desempenhou um papel crucial em manter o humor e a sanidade (principalmente esta parte) em equilíbrio, lembrando-me de que a vida transcende as obrigações acadêmicas.

À Antonio Agripino, que se mostrou um mentor ao longo de todo o meu percurso, não apenas no estágio, mas também em diversas outras atividades relacionadas à microeletrônica. Sua ajuda foi essencial para superar minhas dúvidas e corrigir falhas, e é difícil expressar em palavras o quanto sua orientação foi valiosa em cada etapa.

À professora Georgina, meu sincero agradecimento pela orientação rigorosa e pelos insights valiosos. Você não apenas me ensinou a teoria necessária, mas também como aplicá-la de forma crítica e questionadora. Agradeço por ter exigido o melhor de mim, preparando-me não só para concluir este projeto, mas para enfrentar os desafios que o futuro reserva.

A jornada foi árdua, mas cada obstáculo superado adicionou uma camada de resiliência ao meu caráter. Este ciclo que se fecha foi pontuado tanto por rigor acadêmico quanto por momentos de leveza, e por tudo isso, minha gratidão é imensa.

*“Todo túnel tem uma saída, mesmo que seja para outro túnel.  
(Autoria própria)*

# Lista de ilustrações

Figura 1 – Diagrama arquitetural de blocos de um ADC SAR. . . . .	17
Figura 2 – Circuito de uma chave com bootstrap. . . . .	19
Figura 3 – Exemplo de um circuito de DAC. . . . .	20
Figura 4 – Circuito do latch StrongARM. . . . .	22
Figura 5 – Circuito de lógica SAR para o ADC. . . . .	23
Figura 6 – Circuito de teste do bloco de sample-and-hold . . . . .	25
Figura 7 – Configuração do simulação no ambiente maestro para verificação do funcionamento da amostragem do sinal. . . . .	26
Figura 8 – Circuito de teste do comparador dinâmico. . . . .	27
Figura 9 – Configuração do simulação no ambiente maestro para verificação do funcionamento do comparador dinâmico. . . . .	27
Figura 10 – Circuito de teste para a análise de ruído do comparador. . . . .	28
Figura 11 – Configuração do ambiente de simulação para a análise de ruído do comparador. . . . .	28
Figura 12 – Configuração do ambiente de simulação para a análise de offset do comparador. . . . .	29
Figura 13 – Configuração do ambiente de simulação para a análise de offset do comparador. . . . .	29
Figura 14 – Circuito de teste da lógica de aproximação sucessiva. . . . .	30
Figura 15 – Configuração do simulação no ambiente maestro para verificação do funcionamento do circuito de lógica de aproximações sucessivas. . . . .	30
Figura 16 – Circuito de teste do circuito de gerador de clock para operação assíncrona do ADC SAR. . . . .	31
Figura 17 – Configuração do simulação no ambiente maestro para verificação do funcionamento do circuito de gerador de clock para operação assíncrona do ADC SAR. . . . .	32
Figura 18 – Circuito de teste de descasamento para o DAC. . . . .	32
Figura 19 – Configuração do ambiente de simulação para o teste de descasamento para o DAC. . . . .	33
Figura 20 – Circuito de teste para o ADC SAR. . . . .	33
Figura 21 – Configuração do ambiente de simulação para o ADC SAR. . . . .	34
Figura 22 – Simulação de Sample-and-Hold . . . . .	35
Figura 23 – Simulação de Sample-and-Hold com ampliação da escala de exibição, para melhor detalhamento. . . . .	36
Figura 24 – Simulação do comparador dinâmico. . . . .	36
Figura 25 – Simulação do bloco de lógica de aproximação sucessiva. . . . .	37

Figura 26 – Simulação do bloco de lógica de aproximação sucessiva, com ampliação no momento do <i>End-of-Conversion</i> . . . . .	37
Figura 27 – Simulação do gerador de clock para o conversor. . . . .	37
Figura 28 – Simulação do gerador de clock para o conversor, com ampliação para melhor visualização da largura de pulso de amostragem. . . . .	38
Figura 29 – DFT de 1024 pontos para avaliação das frequências presentes no sinal de saída do DAC. . . . .	38
Figura 30 – Simulação do funcionamento do ADC SAR com todas as etapas de conversão. . . . .	39
Figura 31 – Simulação do funcionamento do ADC SAR com todas as etapas de conversão (escala ampliada para melhor visualização). . . . .	39

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
<b>1.1</b>	<b>Objetivos</b>	<b>11</b>
1.1.1	Objetivo Geral	11
1.1.2	Objetivos Específicos	12
<b>1.2</b>	<b>Método</b>	<b>12</b>
<b>1.3</b>	<b>Estrutura do relatório</b>	<b>12</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>14</b>
<b>2.1</b>	<b>Conversores Analógico-Digitais</b>	<b>14</b>
2.1.1	Métricas de Desempenho	15
2.1.2	ADC de Aproximações Sucessivas ( <i>ADC SAR</i> )	16
<b>3</b>	<b>PROJETO E SIMULAÇÃO</b>	<b>18</b>
<b>3.1</b>	<b>Projeto do ADC SAR</b>	<b>18</b>
3.1.1	Sample-and-Hold (S&H)	18
3.1.1.1	Bootstrapped Switch	18
3.1.2	Conversor Digital-Analógico (DAC)	19
3.1.3	Comparador	21
3.1.4	Lógica SAR	23
3.1.5	Gerador de Clock Interno	24
<b>3.2</b>	<b>Simulações</b>	<b>25</b>
3.2.1	Sample-and-Hold (S&H)	25
3.2.2	Comparador	26
3.2.3	Lógica SAR	29
3.2.4	Gerador de Clock Interno	30
3.2.5	Conversor Digital-Analógico (DAC)	31
3.2.6	Conversor Analógico Digital SAR	31
<b>4</b>	<b>RESULTADOS E DISCUSSÕES</b>	<b>35</b>
<b>4.1</b>	<b>Sample-and-Hold (S&amp;H)</b>	<b>35</b>
<b>4.2</b>	<b>Comparador</b>	<b>35</b>
<b>4.3</b>	<b>Lógica SAR</b>	<b>35</b>
<b>4.4</b>	<b>Gerador de Clock Interno</b>	<b>36</b>
<b>4.5</b>	<b>Conversor Digital-Analógico (DAC)</b>	<b>38</b>
<b>4.6</b>	<b>Conversor Analógico-Digital SAR</b>	<b>38</b>
<b>4.7</b>	<b>Consumo energético do sistema</b>	<b>39</b>

<b>5</b>	<b>CONCLUSÃO</b> . . . . .	<b>41</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>42</b>
	<b>APÊNDICES</b>	<b>43</b>
	<b>APÊNDICE A – CÓDIGO VERILOG-A DO ADC IDEAL DE 10 BITS</b>	<b>45</b>
	<b>APÊNDICE B – CÓDIGO VERILOG-A DO DAC IDEAL DE 10 BITS</b>	<b>48</b>
	<b>APÊNDICE C – CÓDIGO VERILOG-A DO GERADOR DE RAMPA</b>	<b>49</b>

# 1 Introdução

O avanço tecnológico, impulsionado pela demanda por dispositivos portáteis, conectados e de baixo consumo de energia, exige soluções de circuitos e sistemas capazes de atender a esses requisitos rigorosos sem comprometer o desempenho no processamento de dados. Isso se reflete especialmente na área de dispositivos móveis, sensores e sistemas embarcados, onde o balanço entre eficiência energética e *performance* é crucial. (HARPE, 2022) Um exemplo claro de aplicação dessas tecnologias encontra-se no contexto da *Internet das Coisas* (do inglês, *Internet of Things — IoT*), que envolve a comunicação entre dispositivos inteligentes de maneira integrada e sem fio. No âmbito da *IoT*, o sensoriamento remoto, torna-se uma aplicação central, sendo essencial em diversos setores, como automação residencial, monitoramento ambiental, agricultura de precisão, e indústria 4.0. (HONG; LIN; CHIU, 2019) No entanto, a constante comunicação e operação desses dispositivos exigem um consumo energético considerável, o que impõe desafios significativos ao design de circuitos de aquisição e processamento de sinais, onde a eficiência energética e a miniaturização são cada vez mais vitais.

Outro campo que se beneficia diretamente dos circuitos de baixo consumo energético é o da saúde, com o desenvolvimento de dispositivos médicos portáteis e implantáveis. Exemplos incluem monitores cardíacos, sensores de glicose e sistemas de estimulação neural, que devem operar por longos períodos, muitas vezes alimentados por baterias de dimensões reduzidas. (ZHAO et al., 2022) Esse cenário reforça a necessidade de circuitos eficientes, que permitam um consumo mínimo de energia enquanto garantam a precisão e confiabilidade no processamento de sinais vitais. Também fica evidente que a otimização do consumo energético é um fator-chave para prolongar a vida útil desses dispositivos, reduzindo a necessidade de intervenções cirúrgicas para troca de baterias em dispositivos implantáveis, por exemplo. (MAO et al., 2019)

No contexto da *IoT*, o processo de sensoriamento desempenha um papel crucial no funcionamento eficiente dos dispositivos. O sensoriamento envolve a captura de sinais provenientes do ambiente externo — como temperatura, pressão, umidade, ou sinais biométricos no caso de dispositivos médicos. Esses sinais, que são tipicamente analógicos, na maioria dos sistemas, devem ser convertidos para um formato que os sistemas digitais possam processar, o que exige a utilização de conversores analógico-digitais (*ADCs*). Esses conversores são elementos fundamentais no processo de integração de sistemas, uma vez que os sinais capturados pelos sensores analógicos, embora representem as variáveis físicas do ambiente, não são adequados para processamento direto em sistemas digitais, como microcontroladores e processadores de sinal digital.

O processo de conversão analógico-digital envolve a transformação do sinal contínuo, que possui uma infinidade de valores possíveis, em um sinal discreto, formado por uma sequência de bits que pode ser interpretada pelo sistema digital. A precisão dessa conversão é medida pela resolução do conversor, ou seja, o número de bits que ele utiliza para representar o sinal analógico. Conversores com maior resolução são capazes de captar variações mais sutis no sinal, resultando em maior precisão. Entretanto, há um compromisso entre a precisão e o consumo de energia, uma vez que conversores de maior resolução tendem a consumir mais energia, o que representa um desafio na implementação de sistemas *IoT* que dependem de baterias ou outras fontes de energia limitadas. (VERMA et al., 2020)

Além da resolução, outros fatores críticos no design de *ADCs* incluem a velocidade de conversão e a arquitetura do conversor. *ADCs* rápidos são necessários em aplicações que exigem processamento em tempo real, como em sistemas de comunicação sem fio. Entretanto, quanto maior a velocidade, maior o consumo de energia. Esse *trade-off* entre desempenho e eficiência energética exige um design cuidadoso, levando em consideração as características específicas de cada aplicação. (TYAGI; MITTAL; KUMAR, 2023)

Portanto, o desenvolvimento de conversores analógico-digitais eficientes é essencial para garantir o funcionamento de sistemas *IoT* e dispositivos médicos implantáveis, que precisam balancear a necessidade de alta precisão e velocidade com a limitação no consumo de energia. O futuro da eletrônica de baixo consumo está fortemente ligado à inovação nesses componentes, que estão no coração do sensoriamento e do processamento de sinais em ambientes cada vez mais conectados e dinâmicos.

Este trabalho aborda o projeto de conversores analógico-digitais (*ADCs*) de baixo consumo energético a nível de circuito integrado, com foco na arquitetura de aproximações sucessivas. O objetivo principal é desenvolver um *ADC* que atenda à restrição de baixo consumo energético, seguindo o fluxo completo de design de circuitos integrados analógicos — desde as especificações iniciais até a validação de desempenho. A relevância deste estudo reside na crescente demanda por circuitos de alta eficiência e baixo consumo de energia, essenciais para a integração em sistemas mais complexos e aplicações modernas.

## 1.1 Objetivos

### 1.1.1 Objetivo Geral

Realizar o estudo de arquiteturas de conversores analógicos-digitais com enfoque em baixo consumo energético e projetar o circuito com tecnologia de 180 nm.

### 1.1.2 Objetivos Específicos

- Estudar arquiteturas de conversores A/D voltadas para baixo consumo de energia;
- Identificar técnicas para redução do consumo;
- Realizar a modelagem e simulação do conversor a nível de esquemático;
- Validar o desempenho do conversor em termos de consumo, resolução, velocidade e área ocupada do *die*.

## 1.2 Método

O trabalho foi iniciado com uma revisão bibliográfica, conduzida por meio de uma revisão sistemática da literatura. Essa etapa utilizou técnicas de pesquisa e filtragem de resultados com foco em arquiteturas de conversores analógico-digitais de baixo consumo energético. Esse levantamento teórico foi essencial para consolidar conhecimentos sobre a teoria, conceitos, métricas e métodos associados ao projeto de conversores AD.

Em seguida, foi escolhida a arquitetura do conversor a ser estudada, o *ADC SAR (Successive Approximation Register)*. Nessa etapa, realizou-se um estudo detalhado dos parâmetros e métricas de desempenho que caracterizam a eficiência e o funcionamento desse tipo de conversor. Essa análise permitiu compreender a técnica de conversão e identificar os principais desafios envolvidos no projeto. O trabalho avançou com o desenvolvimento e simulação do sistema, bem como com o estudo dos blocos fundamentais que compõem o conversor.

O projeto e as simulações foram executados no software Virtuoso da Cadence, que oferece ferramentas abrangentes para o design e simulação de circuitos, desde o nível de transistor até o nível de blocos funcionais do sistema. A análise dos resultados foi realizada utilizando o *Virtuoso Visualization and Analysis (ViVA)*, que possibilitou a visualização das formas de onda geradas pelo conversor e a medição de parâmetros relevantes, como INL e DNL, entre outros.

## 1.3 Estrutura do relatório

O primeiro capítulo é a Introdução, onde são apresentados os objetivos geral e específicos do trabalho, além da metodologia empregada para alcançá-los. Também é descrita a relevância dos conversores analógico-digitais (ADCs) no contexto de sistemas modernos e a motivação para o estudo de arquiteturas baseadas no método de aproximações sucessivas (SAR). Por fim, a estrutura do relatório é detalhada, fornecendo uma visão geral da organização do conteúdo.

O segundo capítulo, *Fundamentação Teórica*, aborda os conceitos fundamentais relacionados aos conversores AD. Este capítulo começa com uma introdução aos princípios básicos de conversão analógico-digital, incluindo definições de resolução, taxa de amostragem e linearidade. Em seguida, é discutida a arquitetura de aproximações sucessivas (SAR). São apresentados os componentes principais, como comparadores, DACs e lógica de controle, bem como o algoritmo de busca binária que fundamenta o funcionamento do ADC SAR.

O terceiro capítulo, *Projeto e Simulação do Conversor SAR*, descreve o processo de desenvolvimento do ADC. Inicialmente, são apresentados os parâmetros de projeto e as especificações que guiaram o desenvolvimento, como resolução, consumo de energia e frequência de operação. Em seguida, o fluxo de projeto utilizando a plataforma Cadence Virtuoso é detalhado, abordando desde o dimensionamento de transistores até a implementação de blocos funcionais, como comparadores e DACs. Resultados de simulação, como formas de onda e medições de desempenho, são apresentados para validar o funcionamento de cada bloco.

O quarto capítulo, *Resultados e Discussões*, apresenta uma análise detalhada dos resultados obtidos com as simulações e experimentos realizados. São discutidos os aspectos positivos e as limitações do projeto, bem como comparações com referências da literatura e possíveis melhorias.

Finalmente, o quinto capítulo, *Conclusões*, resume os principais resultados obtidos e discute as contribuições do trabalho para o estudo de conversores AD SAR. São destacadas as principais dificuldades enfrentadas, as soluções adotadas e sugestões para trabalhos futuros, como a exploração de técnicas avançadas de redução de consumo energético e aumento de precisão.

## 2 Fundamentação teórica

Neste capítulo, serão apresentados os conceitos fundamentais que embasam o desenvolvimento deste trabalho, focado no projeto de conversores analógico-digitais (ADCs). Inicialmente, serão discutidos os princípios gerais dos conversores AD, destacando os conceitos de amostragem, quantização e codificação digital, que formam a base do funcionamento desses dispositivos. Em seguida, serão introduzidas as principais métricas de desempenho utilizadas para avaliar a qualidade dos conversores, como resolução, velocidade de conversão, consumo de energia, *INL* (*Integral Non-Linearity*) e *DNL* (*Differential Non-Linearity*).

Posteriormente, será detalhada a arquitetura do conversor *ADC SAR* (*Successive Approximation Register*), que é o foco deste trabalho. Essa abordagem incluirá uma descrição do princípio de operação dessa arquitetura, desde a inicialização até o processo iterativo de aproximações sucessivas, bem como a análise de suas vantagens e limitações em termos de eficiência energética e complexidade de implementação.

Em seguida, serão explorados os blocos fundamentais que compõem um *ADC SAR*, como o comparador, o registro de aproximação sucessiva, o conversor digital-analógico (DAC) e o circuito de clock. Será analisada a interação entre esses blocos e sua influência no desempenho geral do conversor, com foco em desafios específicos, como a linearidade do DAC e a velocidade do comparador.

Por fim, será apresentado o fluxo de projeto de circuitos integrados analógicos utilizado para o desenvolvimento do ADC SAR, abrangendo as etapas de especificação, modelagem, projeto em nível de transistor, simulação e validação de desempenho. Este fluxo será descrito de forma a estabelecer as conexões entre os conceitos teóricos discutidos e as ferramentas práticas de projeto, criando uma base sólida para o desenvolvimento das etapas subsequentes deste trabalho.

### 2.1 Conversores Analógico-Digitais

Os conversores analógico-digitais (ADCs) desempenham um papel essencial em sistemas modernos de processamento de sinais, transformando sinais analógicos em códigos digitais para que possam ser manipulados por processadores digitais. Um exemplo prático é o uso em smartphones, onde o ADC converte sinais de áudio em códigos digitais para processamento interno.

O avanço da tecnologia de semicondutores, impulsionado pela miniaturização de transistores, permitiu integrar ADCs em um único chip, reduzindo custos e melhorando

o desempenho. Apesar do domínio do processamento digital, o mundo real permanece analógico, tornando os ADCs indispensáveis como interfaces entre os domínios analógico e digital.

Com o crescimento de aplicações em áreas como redes de sensores e Internet das Coisas (IoT), a demanda por ADCs de alta eficiência, velocidade e baixo consumo de energia continua a aumentar, reforçando sua relevância no avanço de tecnologias futuras.

### 2.1.1 Métricas de Desempenho

Um conversor analógico-digital (ADC) é caracterizado por diversos parâmetros que definem sua precisão, velocidade e desempenho geral. Entre esses parâmetros, destacam-se:

1. **Resolução (N):** Representa o número de bits no código digital de saída do ADC. Um ADC de  $N$  bits pode representar  $2^N$  níveis discretos de tensão, com maior resolução resultando em maior precisão na conversão.
2. **Passo de quantização (LSB):** Conhecido como  $V_{LSB}$ , é o menor incremento de tensão no sinal de entrada que resulta em uma mudança de 1 bit no código de saída. É dado pela relação:

$$V_{LSB} = \frac{V_{REF}}{2^N}, \quad (2.1)$$

onde  $V_{REF}$  é a tensão de referência do ADC.

3. **Taxa de amostragem:** Define o número de conversões realizadas por segundo. É inversamente proporcional ao tempo de conversão e deve ser suficientemente alta para evitar perdas de informações no sinal, de acordo com o teorema de Nyquist.
4. **Tempo de conversão:** Corresponde ao tempo necessário para que o ADC complete uma conversão do sinal de entrada em um código digital.
5. **Faixa dinâmica (Dynamic Range):** Refere-se à relação entre o maior e o menor sinal que o ADC pode processar sem distorções, geralmente expressa em decibéis (dB).
6. **Linearidade (INL e DNL):** A linearidade integral (*Integral Non-Linearity*, INL) e a linearidade diferencial (*Differential Non-Linearity*, DNL) são métricas que avaliam a precisão do ADC em relação à idealidade da curva de transferência. INL mede o desvio absoluto em relação à curva ideal, enquanto DNL avalia a uniformidade entre passos consecutivos de quantização.

Além disso, ADCs podem ser classificados com base em critérios como frequência de amostragem — sendo categorizados como conversores de Nyquist ou de sobreamostragem — ou velocidade de operação, bem como faixa dinâmica, que é a relação entre o maior e o menor sinal que podem processar sem distorções, pela linearidade (avaliada por métricas como INL e DNL) e pelo consumo de energia. Por exemplo, ADCs flash oferecem alta velocidade, mas consomem mais energia, enquanto ADCs SAR podem equilibrar eficiência energética e precisão, dependendo da implementação. A escolha da arquitetura depende dos requisitos da aplicação, como resolução, taxa de amostragem e consumo de energia.

Esses parâmetros e características são cruciais para determinar a adequação de um ADC em sistemas modernos, como dispositivos de Internet das Coisas (IoT), redes de sensores e aplicações de telecomunicações, onde alta precisão e baixo consumo de energia são frequentemente essenciais.

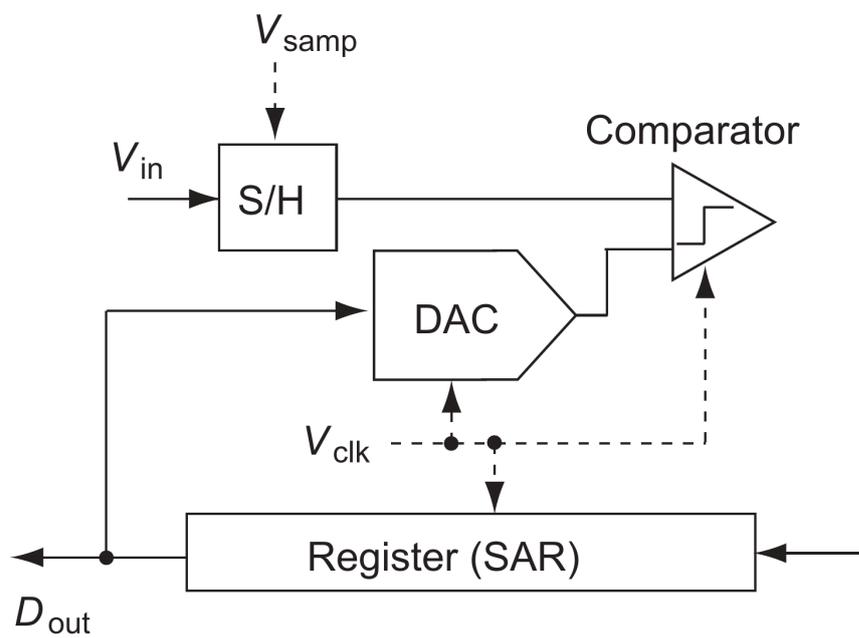
### 2.1.2 ADC de Aproximações Sucessivas (ADC SAR)

O conversor analógico-digital por aproximações sucessivas (SAR ADC) utiliza um algoritmo de busca binária para realizar a conversão de sinais analógicos em códigos digitais. Os blocos fundamentais de um SAR ADC incluem um comparador, um circuito de amostragem e retenção (*Sample-and-Hold*, S&H), um conversor digital-analógico (DAC), uma lógica de controle e registradores. O princípio de funcionamento baseia-se no processo de busca binária, onde a conversão ocorre ao longo de vários ciclos de clock. Inicialmente, o sinal de entrada é amostrado, e o circuito de S&H, frequentemente integrado ao DAC, mantém o valor durante a conversão. A Figura 1 ilustra um diagrama de blocos básico de um ADC SAR.

Em cada etapa, o comparador avalia a diferença entre o sinal de entrada amostrado e a saída do DAC, determinando o valor de cada bit, começando pelo mais significativo (MSB) e progredindo até o menos significativo (LSB). A cada comparação, a diferença entre os valores amostrados e a saída do DAC converge para zero, completando a conversão em  $N$  passos para um SAR ADC de  $N$ -bits.

A lógica de controle pode ser síncrona ou assíncrona. Quando controlado por uma lógica síncrona, o clock interno precisa ser pelo menos  $N$  vezes mais rápido que a frequência de amostragem, o que pode limitar a eficiência em altas taxas de conversão. Em situações que exigem taxas médias ou altas de conversão (na ordem de centenas de MS/s), a lógica assíncrona é preferida, pois elimina a necessidade de um clock de alta velocidade, reduzindo o consumo de energia.

Figura 1 – Diagrama arquitetural de blocos de um ADC SAR.



Fonte: Adaptado de (WAHO, 2019).

## 3 Projeto e simulação

Neste capítulo, serão apresentados o projeto do ADC SAR, detalhado em seus blocos funcionais, e as simulações realizadas para verificar o desempenho e a funcionalidade do conversor.

### 3.1 Projeto do ADC SAR

O projeto do ADC SAR foi realizado com base nos blocos funcionais principais: Sample-and-Hold (S&H), DAC, Comparador, Lógica SAR e Gerador de Clock Interno. O S&H é responsável pela amostragem do sinal de entrada, o DAC gera as referências para as aproximações, o comparador realiza as decisões binárias, a lógica SAR controla o algoritmo de aproximação, e o gerador de clock coordena o funcionamento do sistema. Cada bloco foi projetado e validado individualmente para garantir o desempenho do conversor.

Neste trabalho, o ADC SAR será projetado utilizando a tecnologia CMOS planar de 180 nm, operando com uma tensão de alimentação de 1,8V. O conversor terá uma resolução de 10 bits e terá de operar com uma taxa de amostragem de 10 MS/s. Almeja-se alcançar um *Effective Number of Bits (ENOB)* de 9 bits.

#### 3.1.1 Sample-and-Hold (S&H)

O bloco de *sample-and-hold* captura o valor de um sinal analógico em um instante de tempo e o mantém constante durante um período definido. Em implementações básicas, chaves MOSFET, sejam NMOS, PMOS ou *gates* de transmissão CMOS, são usados. Contudo, essas chaves introduzem não idealidades como:

- Dependência do  $R_{on}$  em relação ao sinal de entrada.
- Injeção de carga no capacitor de saída ao desligar.
- *Feedthrough* de clock devido a capacitâncias parasitas.

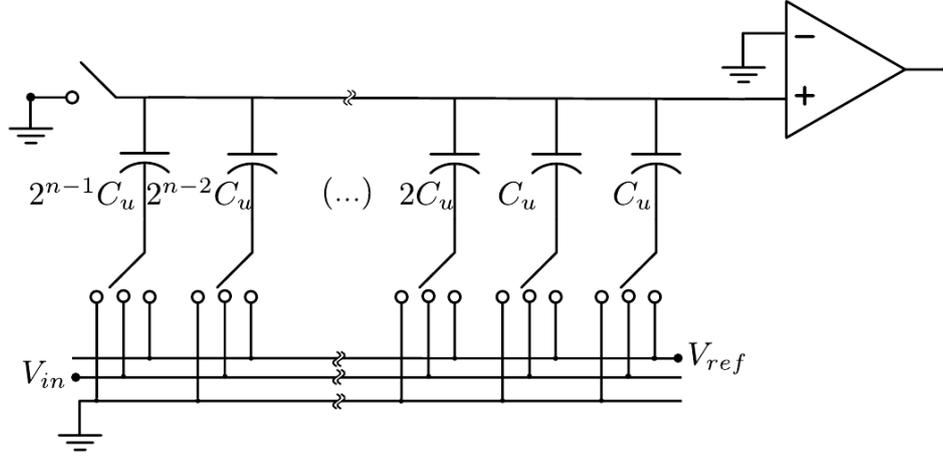
Essas limitações afetam a linearidade, precisão e eficiência energética do circuito, particularmente em tecnologias nanométricas. (RAZAVI, 2015a)

##### 3.1.1.1 Bootstrapped Switch

O *bootstrapped switch* resolve esses problemas ao fixar a tensão  $V_{GS}$  da chave principal durante a fase de amostragem. Isso é alcançado através de um capacitor previamente



Figura 3 – Exemplo de um circuito de DAC.



Fonte: Adaptado de (RICCI, 2020).

$$\sigma_{n,R}^2 = \frac{kT}{C} \quad (3.1)$$

Esse ruído é gerado sempre que ocorre a amostragem do sinal de entrada, independentemente da resistência das chaves. Como o sinal é amostrado no arranjo de capacitores do DAC, a capacitância total  $C_{\text{total}}$  deve ser suficiente para garantir que o ruído permaneça abaixo do nível de ruído de quantização de um conversor de 10 bits. Essa exigência resulta na seguinte condição:

$$C_{\text{total}} \geq \frac{kT}{\left(\frac{\text{LSB}}{\sqrt{12}}\right)^2} \quad (3.2)$$

$$C_{\text{total}} \geq 4,02\text{fF} \quad (3.3)$$

em que LSB é o passo de quantização,  $k$  é a constante de Boltzmann e  $T$  é a temperatura ambiente em Kelvin,  $300\text{K}$ .

Logo, é possível determinar o limite inferior para a capacitância unitária, dada por:

$$C_{\text{unit}} = \frac{C_{\text{total}}}{10} \geq 0,402\text{fF} \quad (3.4)$$

que neste caso, devido ao limite mínimo do PDK de  $180\text{nm}$ , o capacitor unitário terá o valor mínimo de  $5,36\text{fF}$ . Então o DAC com todos os capacitores conectados, terá uma capacitância total igual a  $5,4886\text{pF}$ , se considerado a ponderação de forma binária.

Além de ser projetado para minimizar o consumo de energia e a área do chip, o DAC utilizado no ADC SAR de 10 bits também adota uma configuração diferencial. Essa abordagem diferencial oferece vantagens adicionais em relação à topologia de DAC

de entrada única, como maior rejeição ao ruído de modo comum (CMRR), redução de distorções harmônicas pares e ampliação da faixa dinâmica do sinal. No contexto da etapa de *sample-and-hold*, o DAC diferencial desempenha um papel crucial como elemento de armazenamento e comparação de carga. Sua configuração, que utiliza duas redes de capacitores para sinais diferenciais, oferece vantagens como:

- **Alta Rejeição de Ruído:** Melhora a razão de rejeição ao modo comum (CMRR), reduzindo interferências externas.
- **Minimização de Distorções Harmônicas:** Reduz a distorção de harmônicos pares, garantindo maior linearidade.
- **Amplitude Dupla de Sinal:** Dobra o intervalo efetivo de entrada, aprimorando a relação sinal-ruído (SNR).
- **Robustez a Variações:** É menos sensível a variações de processo e flutuações de tensão.

Na etapa de *sample-and-hold*, o DAC diferencial atua como elemento de armazenamento e comparação de carga, garantindo:

- **Maior Linearidade:** Balanceia as cargas entre os ramos diferenciais.
- **Supressão de Offset:** Compensa erros gerados pelas tensões de modo comum.
- **Eficiência Energética:** Utiliza capacitores binários para reduzir consumo energético.

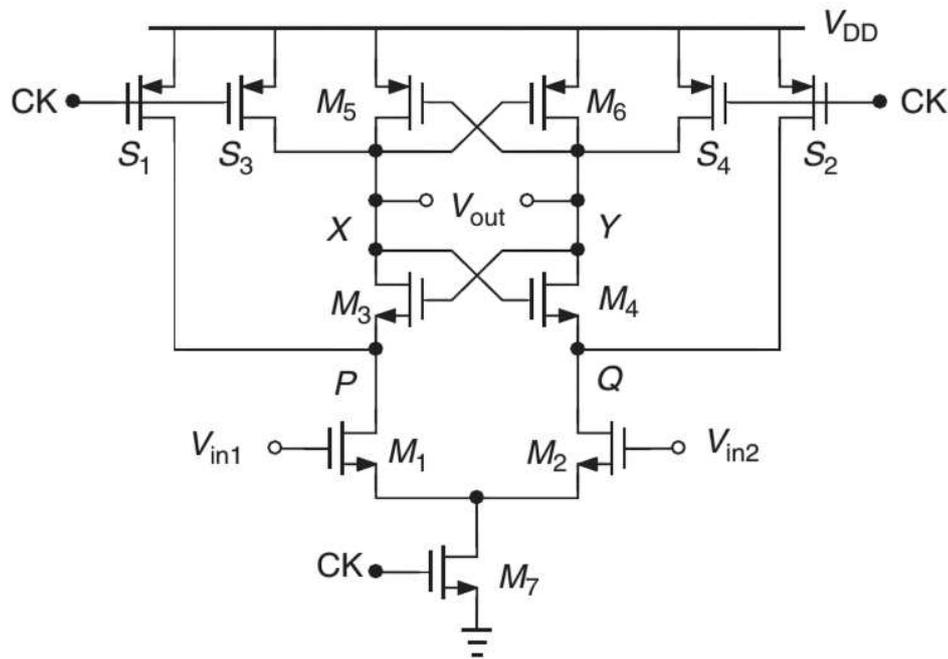
Essas características tornam o DAC diferencial essencial para etapas sensíveis, como o *sample-and-hold* em SAR ADCs.

### 3.1.3 Comparador

O comparador é uma peça-chave em ADCs SAR, desempenhando o papel de decidir o estado lógico de cada bit durante o ciclo de conversão. Entre as várias topologias de comparadores, o latch StrongARM se destaca por sua alta eficiência energética e sensibilidade, sendo amplamente utilizado em sistemas de alta performance. Essa topologia, é reconhecida por consumir zero potência estática, produzir saídas rail-to-rail e possuir um offset de entrada predominantemente influenciado por um único par diferencial. O latch StrongARM é altamente adequado para ADCs SAR devido à sua capacidade de operar em frequências elevadas com baixa dissipação de energia. (RAZAVI, 2015b) O circuito está ilustrado na Figura.

O latch StrongARM é composto por três principais blocos:

Figura 4 – Circuito do latch StrongARM.



Fonte: Adaptado de (RAZAVI, 2015b).

1. **Par diferencial com clock (transistores M1 e M2):** Este bloco gera correntes diferenciais com base na diferença entre os sinais de entrada.
2. **Pares de transistores acoplados de forma cruzada (M3-M4 e M5-M6):** Esses transistores amplificam os sinais diferenciais e garantem uma regeneração positiva para atingir níveis de tensão de saída *rail-to-rail*.
3. **Chaves de pré-carga (S1 a S4):** Estas chaves zeram o estado inicial do circuito antes de cada ciclo de comparação.

Ao projetar o comparador, é essencial considerar:

- **Ruído térmico e offset:** A principal contribuição de ruído e offset vem do par diferencial. Métodos de cancelamento de offset podem incluir ajustes na capacitância dos nós de pré-carga ou o uso de capacitores programáveis.
- **Kickback Noise:** Este efeito é mitigado através do isolamento dos sinais de entrada por meio de switches ou técnicas de neutralização capacitiva.
- **Eficiência energética:** O latch StrongARM consome energia apenas durante as transições, tornando-o ideal para aplicações de baixa potência.



- **Registro SAR:** Armazena o código digital em construção, atualizando-o a cada ciclo de comparação.
- **Interação com o DAC e Comparador:** Durante cada iteração, o DAC ajusta sua saída com base no registro SAR, enquanto o comparador fornece a resposta sobre a proximidade ao valor do sinal analógico.

Esse mecanismo de aproximação sucessiva permite ao ADC SAR realizar conversões precisas, iterando de forma determinística sobre os bits do sinal digital até atingir a resolução desejada.

### 3.1.5 Gerador de Clock Interno

O gerador interno de clock é um componente essencial em ADCs SAR assíncronos, permitindo sincronizar as etapas do processo de conversão de maneira eficiente. Diferentemente da lógica síncrona, que exige um clock global de alta frequência para controlar todas as operações, a lógica assíncrona utiliza um clock global de baixa frequência, complementado por sinais internos gerados dinamicamente. Isso resulta em uma significativa redução no consumo de energia dinâmico do sistema, tornando o design mais eficiente em termos energéticos.

No ADC SAR assíncrono, o gerador interno de clock opera em conjunto com o clock global para produzir os sinais necessários para o funcionamento do sistema. (ALDACHER, 2015) Suas principais operações incluem:

1. **Geração de Pulsos de Amostragem:** Um sinal denominado *clk\_sample* é gerado para acionar a fase de amostragem no circuito de *sample-and-hold*, capturando o sinal analógico de entrada.
2. **Controle da Lógica SAR:** Após a amostragem, o gerador interno produz os sinais que controlam o comparador e ajustam o DAC com base nos resultados das comparações realizadas.
3. **Sinal de Pronto (*Ready*):** O comparador, ao concluir uma comparação, gera um pulso que aciona o gerador para iniciar a próxima iteração.
4. **Sinal de Fim de Conversão (*End-of-Conversion - EOC*):** Após todas as  $N$  iterações necessárias para gerar os bits digitais, o gerador emite um pulso indicando que o código digital está pronto.

Uma das principais vantagens da lógica assíncrona é a eficiência energética proporcionada por sua estrutura de controle. O uso de um clock global, não necessariamente de

alta frequência, reduz o consumo energético dinâmico ao minimizar as transições de estado em altas velocidades. Além disso, como os tempos de ciclo são ajustados dinamicamente com base no comportamento do comparador e na estabilização do DAC, não há desperdício de energia devido a ciclos desnecessários. Isso também permite que os blocos do ADC operem apenas quando necessário, evitando a alimentação contínua de componentes que não estão sendo utilizados.

## 3.2 Simulações

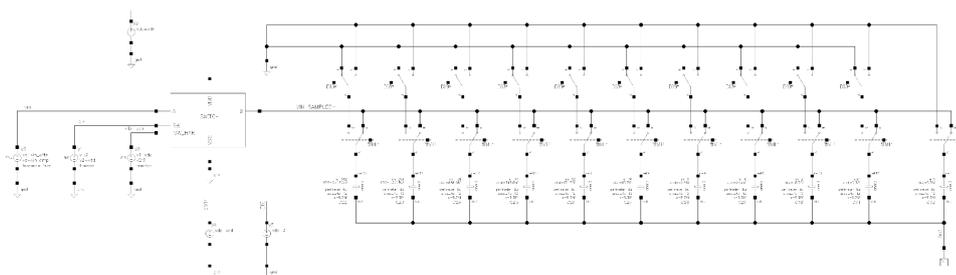
As simulações realizadas neste trabalho têm como objetivo validar e caracterizar o desempenho de cada bloco funcional do ADC SAR, garantindo que o conversor como um todo atenda às especificações de projeto. Para isso, cada componente foi projetado, simulado e analisado individualmente. Também foram implementados blocos auxiliares como um DAC ideal (ALDACHER, 2015), um ADC ideal e um gerador de rampa, sendo esses dois últimos retirados de RAKs da Cadence (Cadence Design Systems, Inc., 2023), todos desenvolvidos em Verilog-A para apoiar as simulações do projeto. Os detalhes e as implementações desses componentes são apresentados no apêndice do documento.

### 3.2.1 Sample-and-Hold (S&H)

O bloco de *Sample-and-Hold* é responsável por capturar e manter o valor do sinal de entrada analógico durante o processo de conversão. As simulações avaliaram a capacidade de amostragem da chave e a estabilidade do sinal durante o período de *hold*.

Para realizar o teste, foi montado um circuito de teste como mostrado na Figura 6. O circuito de teste consiste no bloco de *bootstrapped switch*, seguido de um circuito de DAC capacitivo, que servirá como carga capacitiva da chave, para ser observado como o sinal amostrado é propagado entre os capacitores do DAC. Vale ressaltar que todos os capacitores estão conectados a chave, logo será a soma dos capacitores que servirá como carga totalizando  $4,02fF$ .

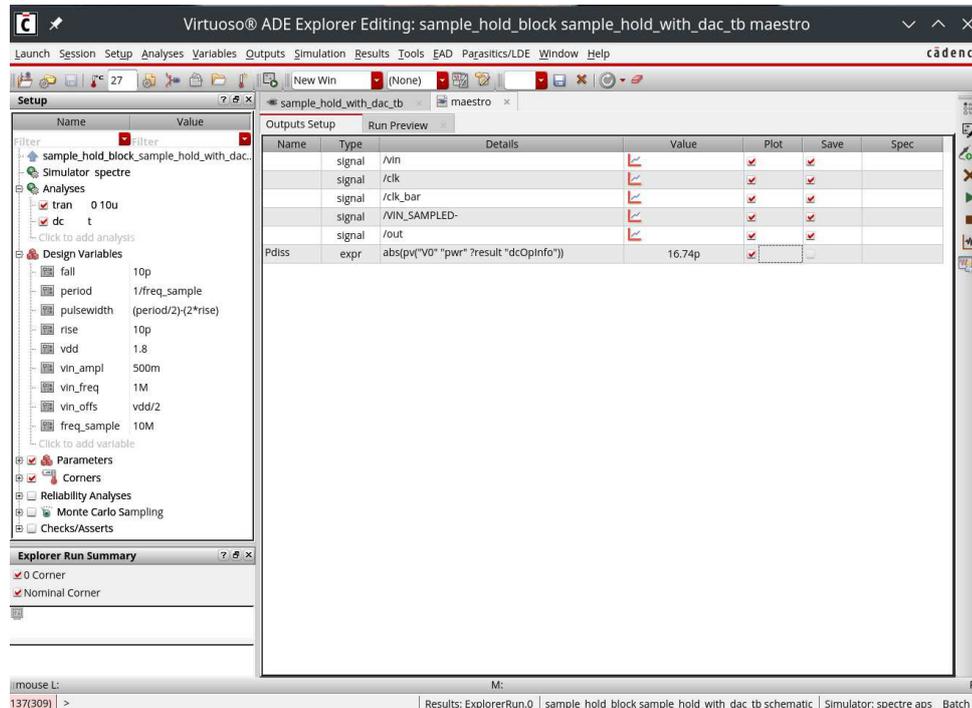
Figura 6 – Circuito de teste do bloco de sample-and-hold



Fonte: [Autoria própria](#).

O ambiente de simulação do maestro do circuito de Sample-and-Hold, foi configurado como mostra a Figura 7. Será medido com essa simulação o erro de amostragem e a potência dissipada por esse bloco.

Figura 7 – Configuração do simulação no ambiente maestro para verificação do funcionamento da amostragem do sinal.



Fonte: [Autoria própria](#).

### 3.2.2 Comparador

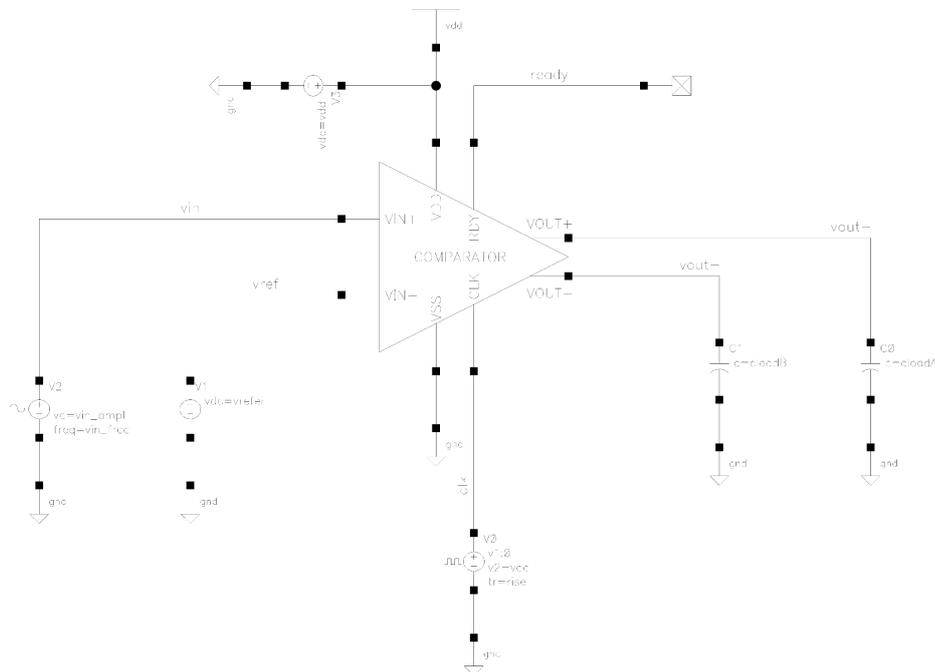
O comparador é responsável por comparar o sinal amostrado no S&H com o sinal gerado pelo DAC, determinando os bits do código digital. As simulações focaram no atraso de propagação, ruído e *offset*.

O circuito de teste do comparador foi configurado como mostrado na Figura 8, tendo como entradas um sinal senoidal de 250mV com offset de  $VDD/2 = 900mV$  e frequência de 2MHz na entrada não inversora do comparador e uma tensão DC de 400mV. Um sinal de clock com frequência de 10MHz.

Para o teste de funcionamento do comparador, foi configurado o ambiente de simulação do maestro como mostrado na Figura 9.

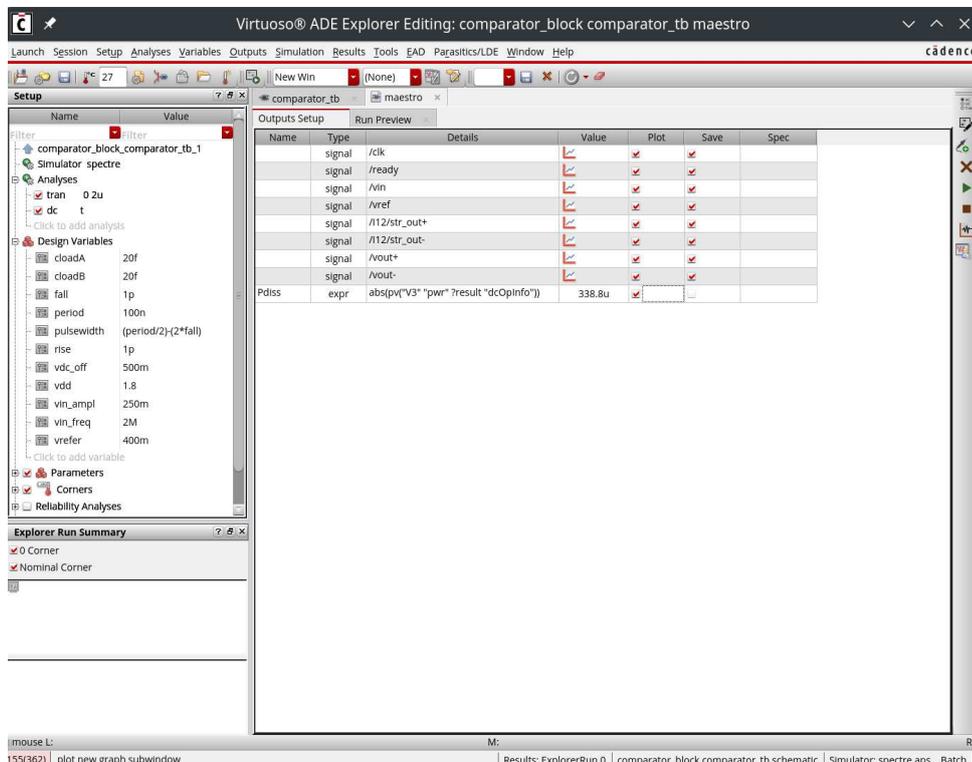
Também, a análise de ruído e offset foi realizada para garantir a precisão necessária. Inicialmente, conduziu-se uma análise de ruído transitório utilizando uma entrada DC para simular vários ciclos de clock e determinar a variação na saída do comparador. Complementarmente, realizou-se uma análise de ruído de estado permanente periódico (PSS) com uma pequena tensão DC, utilizando análise pnoise para avaliar o efeito de ruído

Figura 8 – Circuito de teste do comparador dinâmico.



Fonte: [Autoria própria.](#)

Figura 9 – Configuração do simulação no ambiente maestro para verificação do funcionamento do comparador dinâmico.

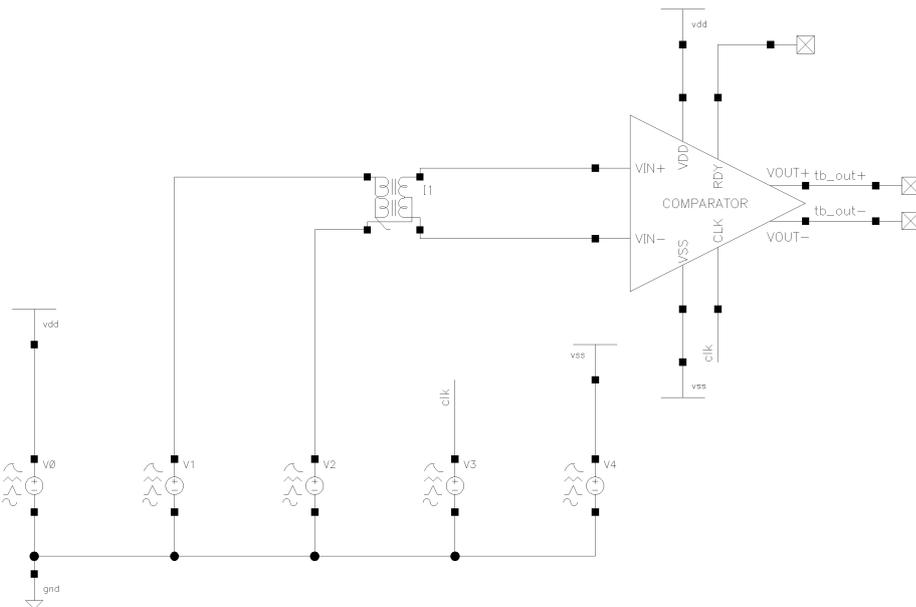


Fonte: [Autoria própria.](#)

na performance do comparador. A caracterização do offset foi feita aplicando uma rampa de tensão e observando a mudança de estado na saída do comparador, fornecendo uma

medida da tensão de offset. As Figuras 10 e 11 mostram o circuito de teste e a configuração do ambiente de simulação para a análise de ruído do comparador, respectivamente. Para a análise de offset, as Figuras 12 e 13 ilustram o circuito de teste e a configuração do ambiente de simulação, respectivamente, destacando as metodologias aplicadas para garantir a precisão das medições.

Figura 10 – Circuito de teste para a análise de ruído do comparador.



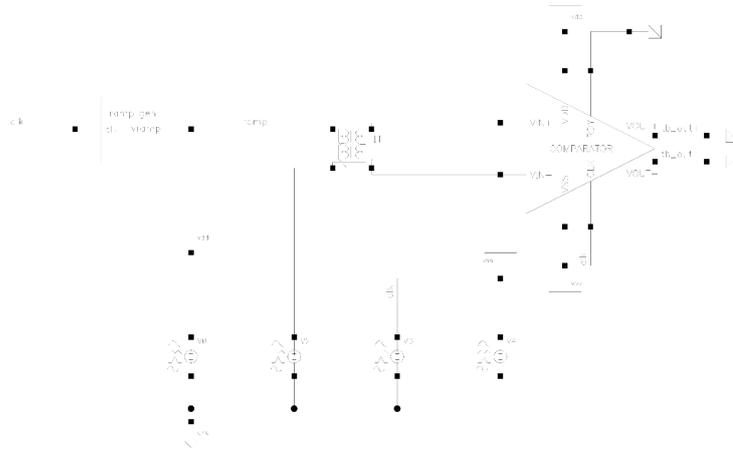
Fonte: [Autoria própria.](#)

Figura 11 – Configuração do ambiente de simulação para a análise de ruído do comparador.

Name	Type	Details	Value	Plot	Save	Spec
PSS	expr	(vtime("pss" "rb_out") - ...				
clk_transient	expr	v("clk" ?result "pss_tran")				
non_invertingoutput_transient	expr	v("rb_out" ?result "pss_...				
str_out_plus_transient	expr	v("r0str_out" ?result "...				
str_out_minus_transient	expr	v("r0str_out" ?result "...				
cross_time	expr	(cross(vtime("pss" "r0s...				
time_pss	expr	(vtime("pss" "r0str_out...				
total_noise_output	expr	valueymax(sqrt(integ...	86.35m			
noise_density	expr	getData("/out" ?result "p...				
input_referred_noise	expr	valueymax(sqrt(integ...	863.5u			

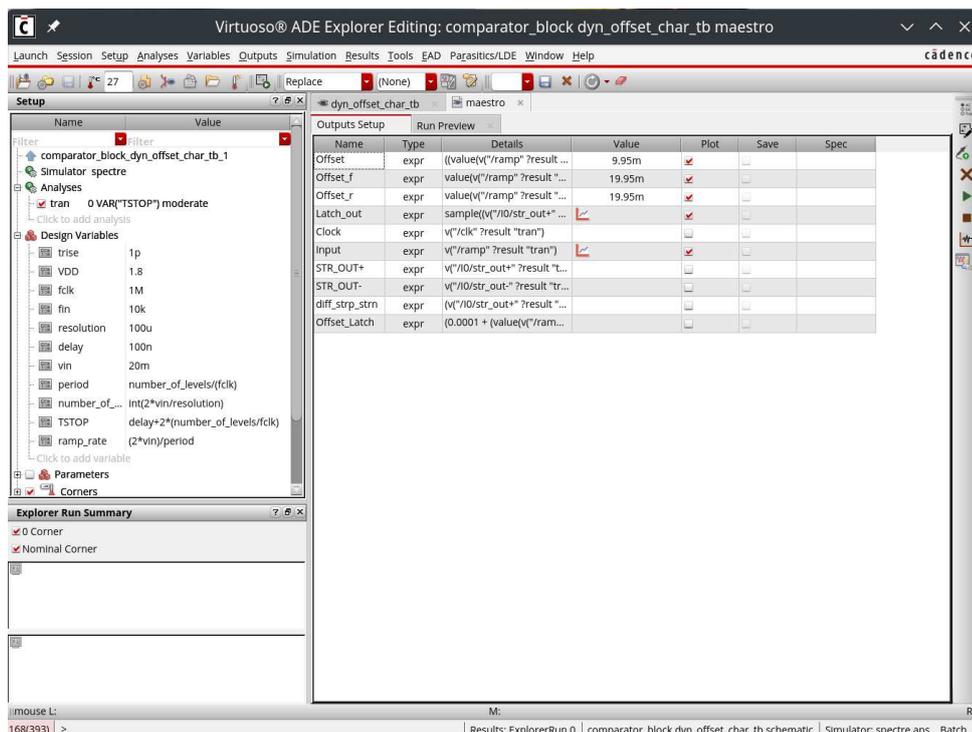
Fonte: [Autoria própria.](#)

Figura 12 – Configuração do ambiente de simulação para a análise de offset do comparador.



Fonte: [Autoria própria.](#)

Figura 13 – Configuração do ambiente de simulação para a análise de offset do comparador.



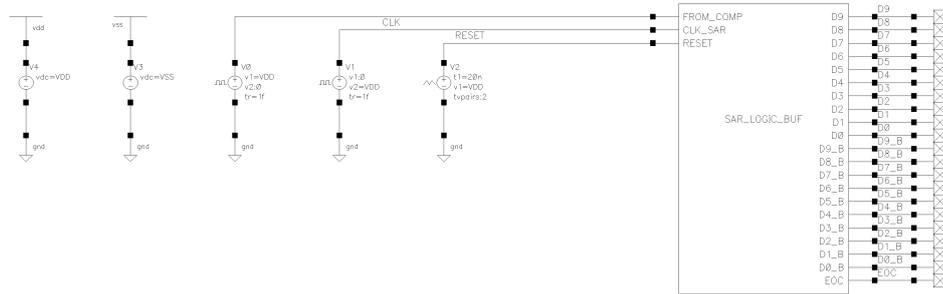
Fonte: [Autoria própria.](#)

### 3.2.3 Lógica SAR

A lógica SAR controla o processo de conversão, implementando o algoritmo de busca binária. Este bloco foi simulado no nível digital para validar o correto sequenciamento das operações e, em integração com o sistema, para garantir a sincronização com os demais blocos.

Para a simulação do bloco de lógica de aproximação sucessivas, foi configurado o circuito de teste mostrado na Figura 14.

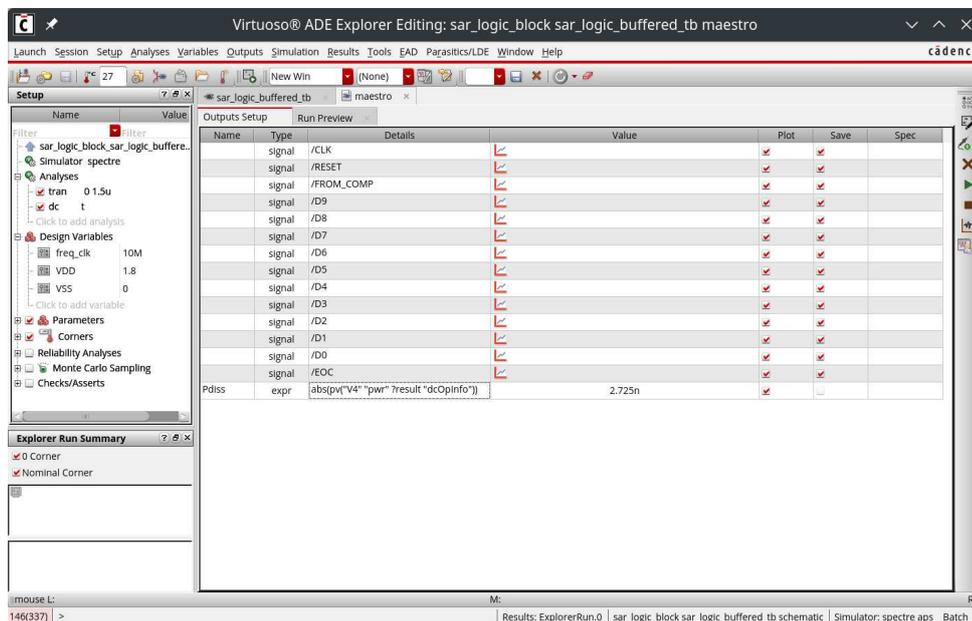
Figura 14 – Circuito de teste da lógica de aproximação sucessiva.



Fonte: [Autoria própria.](#)

Então, para realizar o teste de funcionamento do comparador, foi realizado a configuração de simulação no maestro, como mostra a Figura 15, com frequência de clock de entrada do bloco de 10MHz. Cada borda de clock funciona com que a lógica realize uma operação de aproximação, até que o sinal de EOC seja gatilhado, indicando o fim de conversão do valor analógico para digital.

Figura 15 – Configuração do simulação no ambiente maestro para verificação do funcionamento do circuito de lógica de aproximações sucessivas.



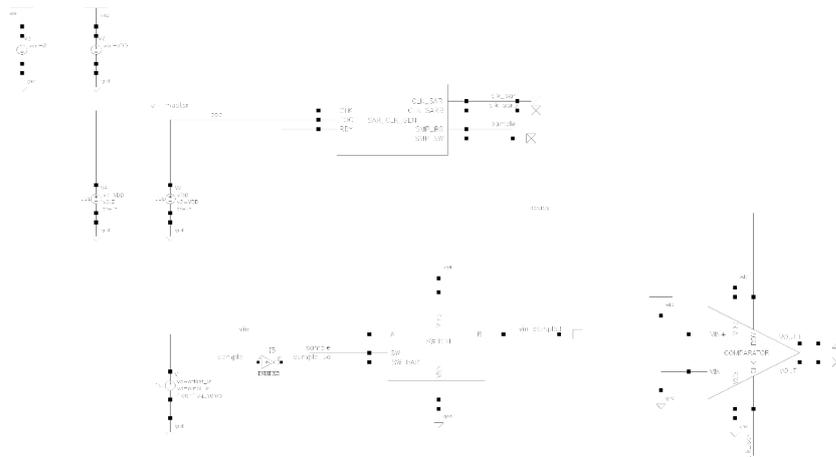
Fonte: [Autoria própria.](#)

### 3.2.4 Gerador de Clock Interno

O gerador de clock interno é responsável por coordenar as operações do ADC SAR de maneira assíncrona. As simulações verificaram a correta geração dos pulsos de clock necessários para o funcionamento do sistema.

Para o teste de funcionamento do gerador de clock interno, o circuito de teste foi configurado usando o bloco do gerador, assim como uma chave de amostragem, que é uma das entradas do DAC, como também o comparador, que fornecerá os pulsos de `ready` necessários, para sinalizar um final de comparação e o gerador prosseguir com o próximo pulso de `clk_sar` que representa o pulso de conversão que será aplicado como entrada do bloco de lógica SAR. O circuito está representado na Figura 16.

Figura 16 – Circuito de teste do circuito de gerador de clock para operação assíncrona do ADC SAR.



Fonte: [Autoria própria](#).

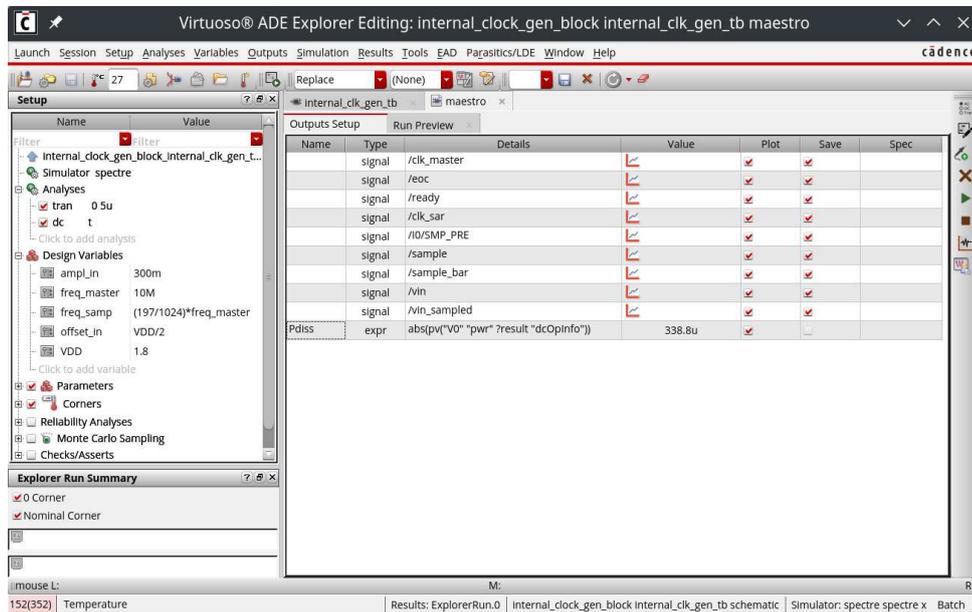
Para a verificação do funcionamento, foi realizada a configuração do maestro com a frequência de clock externa ao que seria o circuito do ADC de  $freq = 10\text{MHz}$ . Também foi utilizado como entrada da chave de amostragem um sinal senoidal com uma frequência de  $(197/1024) \cdot freq$ , amplitude de  $300\text{mV}$  e um *offset* DC de  $500\text{mV}$ . Tal configuração é ilustrada na Figura 17.

### 3.2.5 Conversor Digital-Analógico (DAC)

O DAC desempenha um papel crucial no funcionamento do ADC SAR, pois gera tensões de referência para comparação durante as aproximações sucessivas. Simulações no nível de transistor foram realizadas para verificar a precisão das tensões de saída em relação ao código de entrada.

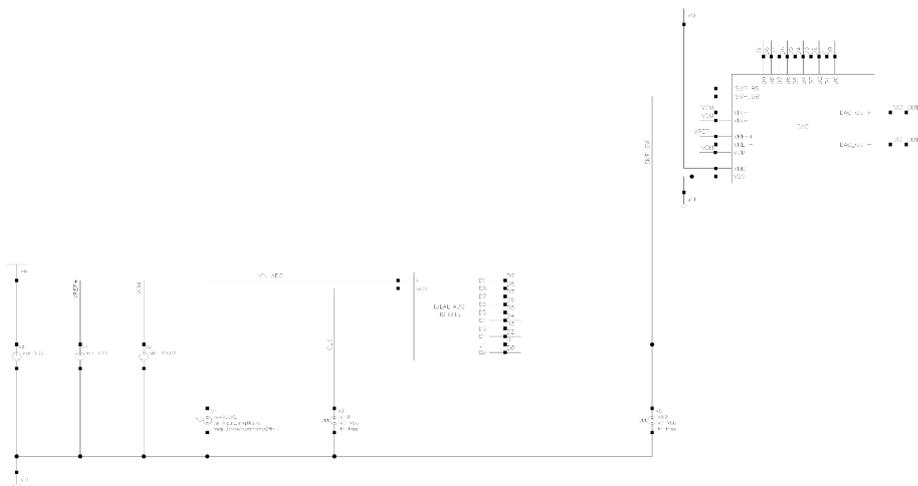
### 3.2.6 Conversor Analógico Digital SAR

Figura 17 – Configuração do simulação no ambiente maestro para verificação do funcionamento do circuito de gerador de clock para operação assíncrona do ADC SAR.



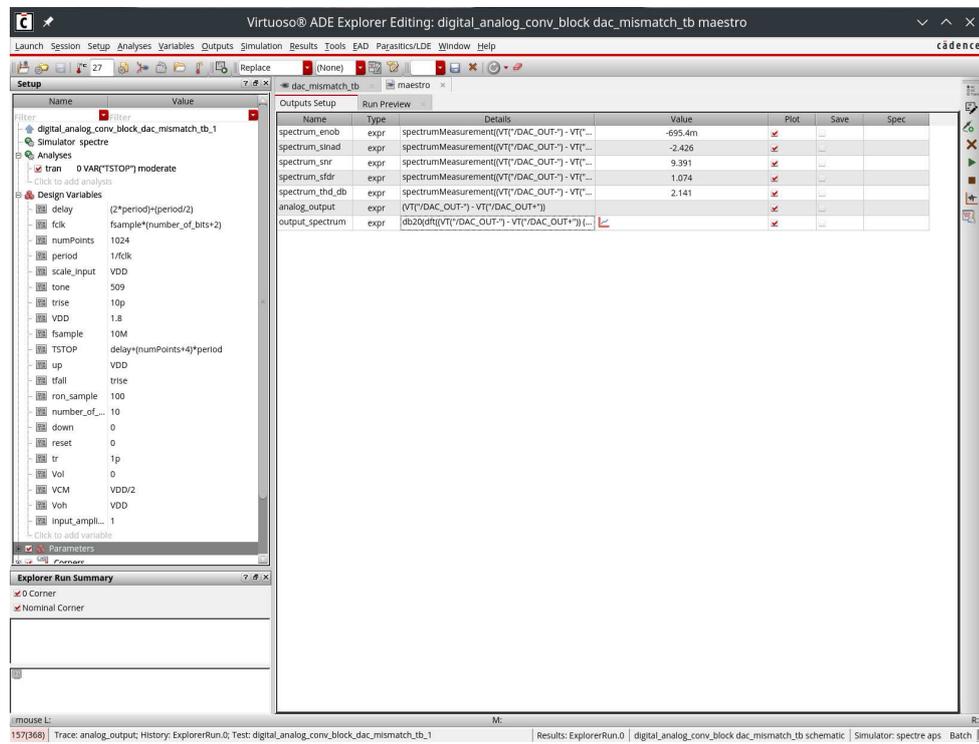
Fonte: [Auria própria.](#)

Figura 18 – Circuito de teste de descasamento para o DAC.



Fonte: [Auria própria.](#)

Figura 19 – Configuração do ambiente de simulação para o teste de descasamento para o DAC.



Fonte: [Auria própria](#).

Figura 20 – Circuito de teste para o ADC SAR.

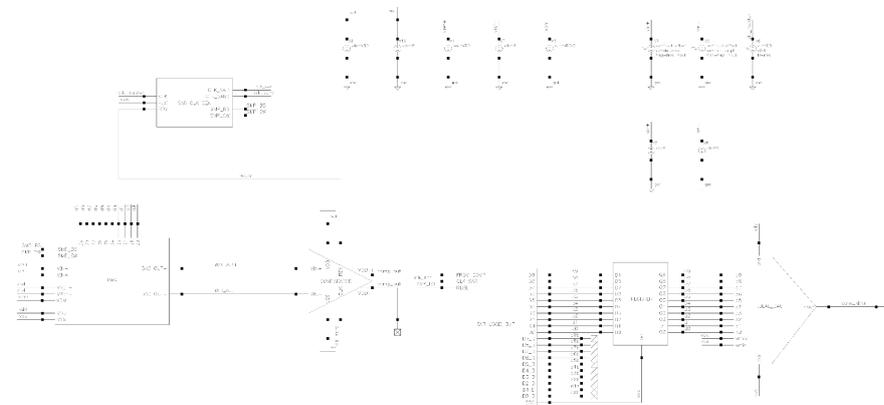
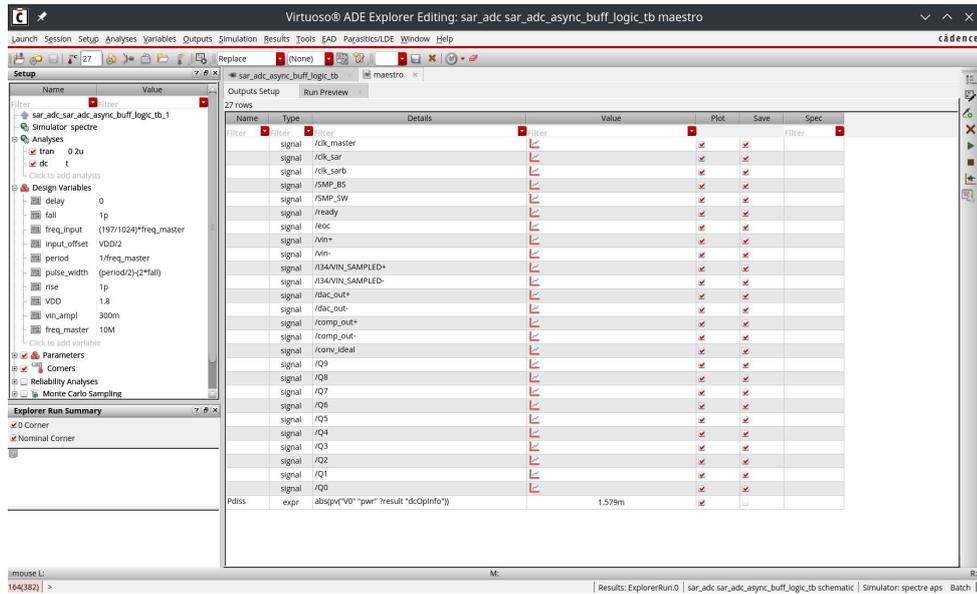


Figura 21 – Configuração do ambiente de simulação para o ADC SAR.



Fonte: Aatoria própria.

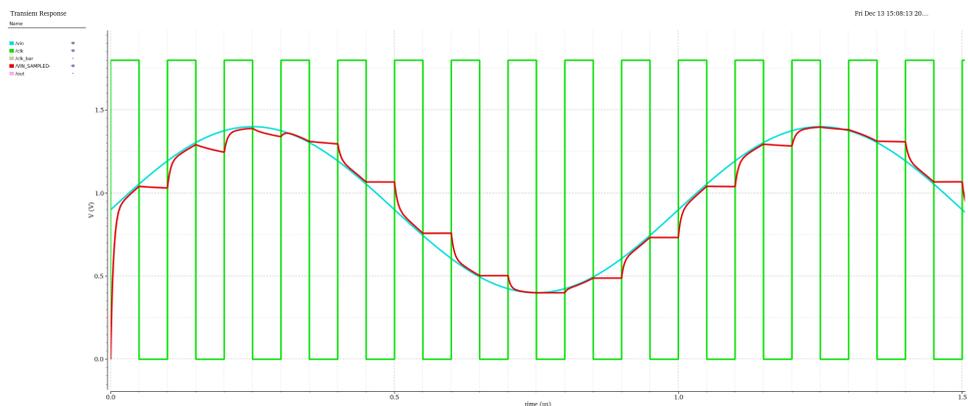
## 4 Resultados e Discussões

Este capítulo apresenta os resultados obtidos com as simulações realizadas para o ADC SAR, detalhando o comportamento e desempenho de cada bloco funcional, com base nas configurações descritas na seção anterior.

### 4.1 Sample-and-Hold (S&H)

As simulações do bloco de Sample-and-Hold (S&H) demonstraram a capacidade do circuito em capturar e manter o sinal de entrada analógico durante o processo de conversão. Os resultados indicam que o uso de switches bootstrapped minimizou os efeitos de injeção de carga. As formas de onda estão apresentadas na Figura 22 e 23, com ampliação da imagem para melhor análise do valor das tensões. É possível observar que o sinal amostrado não tem valores iguais na etapa de rastreamento, tendo um valor menor que o sinal de entrada, devido a queda de tensão pela capacitância dos transistores e a carga capacitiva.

Figura 22 – Simulação de Sample-and-Hold



Fonte: [Autoria própria](#).

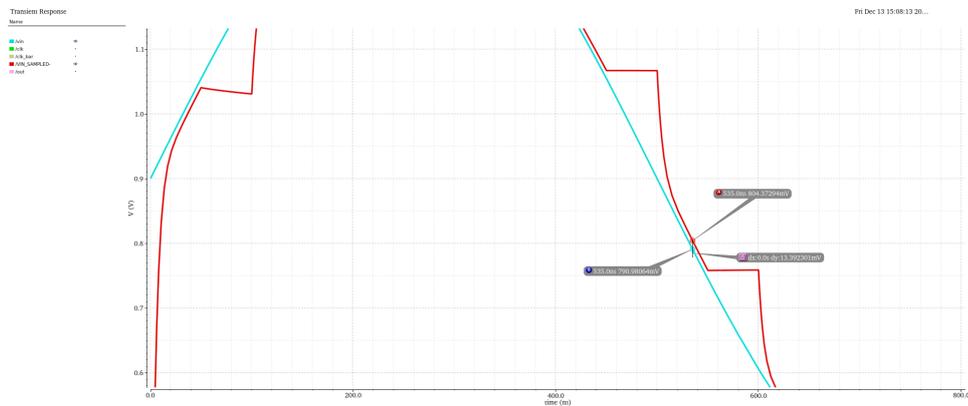
### 4.2 Comparador

O comparador StrongARM operou conforme esperado, determinando de forma eficiente os estados lógicos do sinal de saída em cada ciclo de conversão. A Figura 24 mostra o comportamento dinâmico do comparador.

### 4.3 Lógica SAR

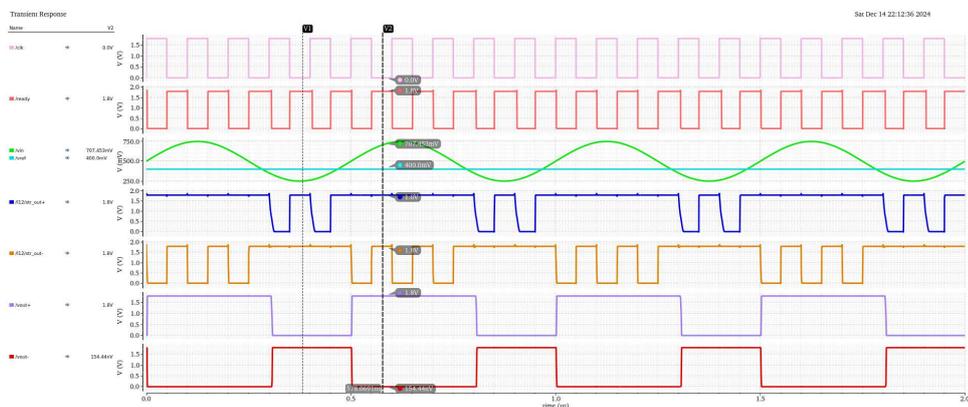
A lógica SAR foi simulada de forma integrada e isolada. O circuito demonstrou operação adequada com o gerador de clock interno e o DAC, permitindo a convergência

Figura 23 – Simulação de Sample-and-Hold com ampliação da escala de exibição, para melhor detalhamento.



Fonte: Autoria própria.

Figura 24 – Simulação do comparador dinâmico.



Fonte: Autoria própria.

precisa dos valores binários em todas as iterações. O sequenciamento das operações de aproximação sucessiva atendeu às especificações do projeto. As Figuras 25 e 26 mostram as formas de onda para o funcionamento da lógica SAR. É possível observar que no último passo de conversão, o sinal de  $EoC$  tem sua borda de subida acontecendo antes da borda de descida do bit menos significativo  $D0$ .

## 4.4 Gerador de Clock Interno

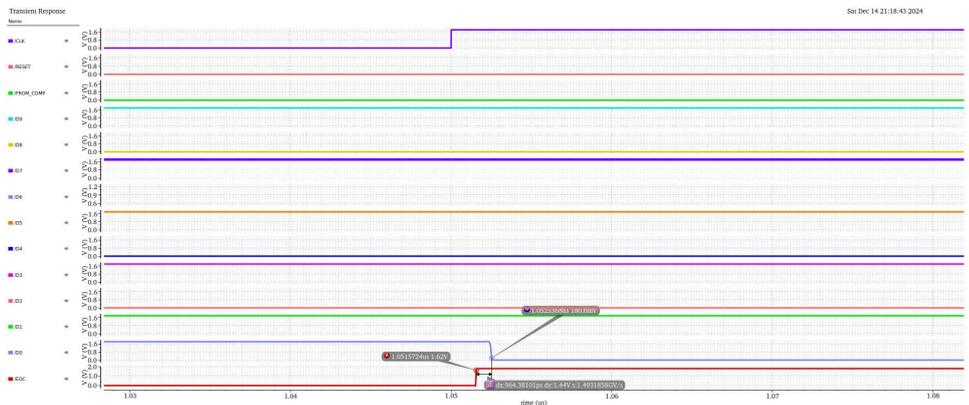
O gerador de clock interno coordenou as operações de conversão de forma assíncrona, fornecendo pulsos consistentes ao longo do processo. Os sinais de controle gerados foram validados em simulação, mostrando transições adequadas e compatíveis com as demandas da lógica SAR e do comparador. A Figura 27 mostra a simulação geral do gerador de clock para o conversor, enquanto a Figura 28 apresenta uma ampliação para melhor visualização da largura de pulso de amostragem, destacando detalhes das transições de clock. É possível observar que a largura do pulso de amostragem é de  $2,81ns$ , para fornecer tempo necessário

Figura 25 – Simulação do bloco de lógica de aproximação sucessiva.



Fonte: Autoria própria.

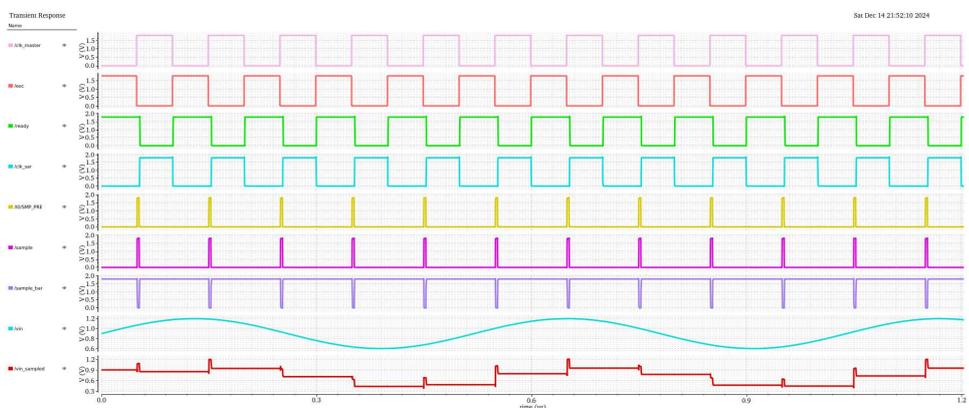
Figura 26 – Simulação do bloco de lógica de aproximação sucessiva, com ampliação no momento do *End-of-Conversion*.



Fonte: Autoria própria.

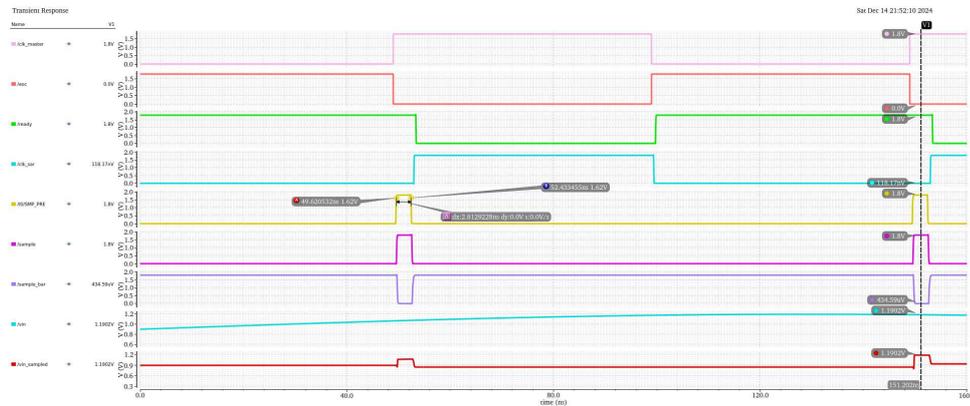
para os capacitores de amostragem serem carregados.

Figura 27 – Simulação do gerador de clock para o conversor.



Fonte: Autoria própria.

Figura 28 – Simulação do gerador de clock para o conversor, com ampliação para melhor visualização da largura de pulso de amostragem.

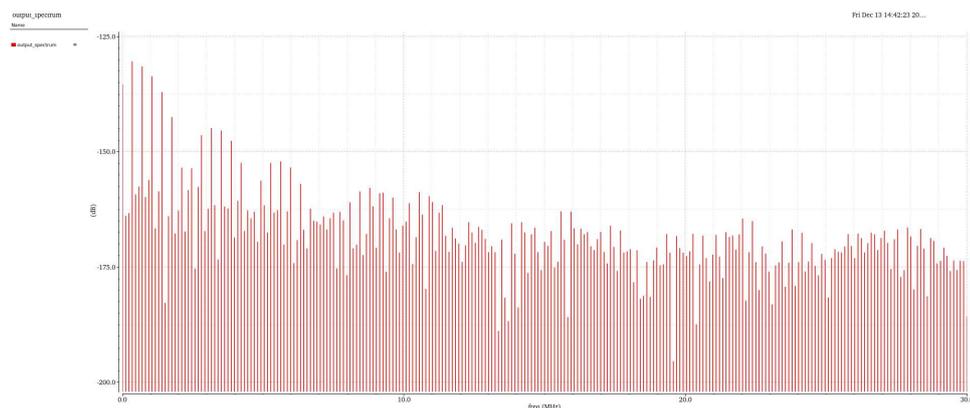


Fonte: [Autoria própria.](#)

## 4.5 Conversor Digital-Analógico (DAC)

O DAC foi avaliado quanto ao descasamento dos capacitores, ENOB, SINAD, SNR, SNDR e THD. Infelizmente o DAC não funcionou apropriadamente, demonstrando valores não satisfatórios, que significa que não satisfaz os critérios de projeto do ADC. A Figura 29 mostra o uma DFT de 1024 pontos do sinal de saída do DAC. Como o circuito não funciona de maneira correta, o gráfico da DFT tem valores que podem ser considerados desprezíveis mesmo para a frequência de operação do circuito.

Figura 29 – DFT de 1024 pontos para avaliação das frequências presentes no sinal de saída do DAC.



Fonte: [Autoria própria.](#)

## 4.6 Conversor Analógico-Digital SAR

Na simulação do funcionamento do Conversor Analógico-Digital SAR, observou-se que o dispositivo não funcionou conforme o esperado. As Figuras 30 e 31 ilustram o teste de todas as etapas de conversão, onde foram identificadas falhas significativas que comprometem a operação adequada do conversor. As análises indicam a necessidade de



Tabela 1 – Potência consumida por cada bloco do ADC SAR

<b>Bloco</b>	<b>Potência (W)</b>
Sample-and-Hold	16,74p
Comparador	338,8 $\mu$
Lógica SAR	2,725n
Gerador de Clock	338,8 $\mu$
DAC	901,397 $\mu$

Fonte: [Autoria própria](#).

## 5 Conclusão

O projeto e as simulações do ADC SAR em tecnologia de 180 nm revelaram desafios significativos. Embora os blocos funcionais tenham demonstrado operação parcial, o desempenho do conversor como um todo foi comprometido. O principal obstáculo identificado foi o funcionamento inadequado do DAC, que comprometeu a integração e o comportamento geral do ADC. Além disso, dificuldades no processo de design dos blocos e na utilização do PDK de 180 nm dificultaram o avanço esperado no projeto.

Os resultados destacam a necessidade de trabalhos futuros focados na correção dos problemas de operação do ADC, com ênfase na integração efetiva dos blocos funcionais para formar um sistema completo. Além disso, recomenda-se o desenvolvimento do design físico ou leiaute do projeto, com vistas a melhorar a robustez e o desempenho do conversor.

# Referências

- ALDACHER, M. *Design of a Low-Power Asynchronous SAR ADC in 45 nm CMOS Technology*. Dissertação (Mestrado) — San Jose State University, 2015. Explores the design and implementation of an 8-bit asynchronous SAR ADC, focusing on low power consumption and efficiency. Citado 3 vezes nas páginas 23, 24 e 25.
- Cadence Design Systems, Inc. *ADC Verification: Rapid Adoption Kit (RAK)*. Product Version: IC23.1/IC6.1.8, SPECTRE 23.1/21.1, 2023. Accessed: 2024-12-05. Disponível em: <<https://support.cadence.com>>. Citado na página 25.
- HARPE, P. Low-power sar adcs: Basic techniques and trends. *IEEE Open Journal of the Solid-State Circuits Society*, v. 2, p. 73–81, 2022. ISSN 2644-1349. Citado na página 10.
- HONG, H.-C.; LIN, L.-Y.; CHIU, Y. Design of a 0.20–0.25-v, sub-nw, rail-to-rail, 10-bit sar adc for self-sustainable iot applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, v. 66, n. 5, p. 1840–1852, maio 2019. ISSN 1558-0806. Citado na página 10.
- MAO, W. et al. A low power 12-bit 1-ks/s sar adc for biomedical signal processing. *IEEE Transactions on Circuits and Systems I: Regular Papers*, v. 66, n. 2, p. 477–488, fev. 2019. ISSN 1558-0806. Citado na página 10.
- RAZAVI, B. The bootstrapped switch [a circuit for all seasons]. *IEEE Solid-State Circuits Magazine*, v. 7, n. 3, p. 12–15, 2015. ISSN 1943-0590. Citado 2 vezes nas páginas 18 e 19.
- RAZAVI, B. The strongarm latch [a circuit for all seasons]. *IEEE Solid-State Circuits Magazine*, v. 7, n. 2, p. 12–17, 2015. ISSN 1943-0590. Citado 2 vezes nas páginas 21 e 22.
- RICCI, L. *Design of a 12-bit 200-MSps SAR Analog-to-Digital converter*. Dissertação (Degree Project Electrical Engineering, Second Cycle) — KTH Royal Institute of Technology, School of Electrical Engineering and Computer Science, Stockholm, Sweden, 2020. Citado 2 vezes nas páginas 19 e 20.
- TYAGI, M.; MITTAL, P.; KUMAR, P. Design of 8-bit low power sar adc in 45 nm for biomedical implants. *Physica Scripta*, IOP Publishing, v. 98, n. 11, p. 116101, out. 2023. Citado na página 11.
- VERMA, D. et al. A design of 8 fj/conversion-step 10-bit 8ms/s low power asynchronous sar adc for iee 802.15.1 iot sensor based applications. *IEEE Access*, v. 8, p. 85869–85879, 2020. ISSN 2169-3536. Citado na página 11.
- WAHO, T. *Introduction to Analog-to-Digital Converters: Principles and Circuit Implementation*. Denmark: River Publishers, 2019. (River Publishers Series in Circuits and Systems). ISBN 978-87-7022-102-3. Disponível em: <<https://www.riverpublishers.com/>>. Citado na página 17.
- ZHAO, X. et al. A 0.6-v 94-nw 10-bit 200-ks/s single-ended sar adc for implantable biosensor applications. *IEEE Sensors Journal*, v. 22, n. 18, p. 17904–17913, set. 2022. ISSN 1558-1748. Citado na página 10.

# Apêndices

—

# APÊNDICE A – Código Verilog-A do ADC ideal de 10 bits

```

1 // VerilogA for ideal_blocks, adc_10bits, veriloga
2
3 'include "constants.vams"
4 'include "disciplines.vams"
5
6 //-----
7 // adc_10bit
8 //
9 // - 10 bit analog to digital converter
10 //
11 // vin:    [V,A]
12 // vclk:   [V,A]
13 // D9..D0: data output terminals [V,A]
14 //
15 // INSTANCE parameters
16 //     mismatch_fact = maximum mismatch as a percentage of the
17 //     average value []
18 //     vlogic_high   = [V]
19 //     vlogic_low    = [V]
20 //     vtrans_clk    = clk high to low transition voltage [V]
21 //     vref          = voltage that voltage is done with respect
22 //     to [V]
23 //     tdel, trise, tfall = {usual} [s]
24 //
25 // MODEL parameters
26 //     {none}
27 //
28 // This ADC comprises 10 comparators. An input voltage is
29 // compared to
30 // half the reference voltage. If the input exceeds it, bit 9 is
31 // set and half
32 // the reference voltage is subtracted. If not, bit 9 is assigned
33 // zero
34 // and no voltage is subtracted from the input. Bit 8 is found by
35 // doing an
36 // equivalent operation comparing double the adjusted input
37 // voltage coming
38 // from the first comparator with half the reference voltage.
39 // Similarly
40 // all the other bits are found.
41 //
42 // Mismatch effects in the comparator reference voltages can be
43 // modeled
44 // setting 'mismatch' to a non-zero value. The maximum mismatch
45 // on a
46 // comparator's reference voltage is then +/- 'mismatch' percent
47 // of

```

```

37 // that voltage's nominal value.
38 //
39
40 module adc_10bit(D9, D8, D7, D6, D5, D4, D3, D2, D1, D0, vin,
    vclk);
41
42 output D9, D8, D7, D6, D5, D4, D3, D2, D1, D0 ;
43 input vin, vclk ;
44
45 electrical D9, D8, D7, D6, D5, D4, D3, D2, D1, D0, vin, vclk;
46
47 parameter real trise = 0 from [0:inf);
48 parameter real tfall = 0 from [0:inf);
49 parameter real tdel = 0 from [0:inf);
50 parameter real mismatch_fact = 0 from [0:inf);
51 parameter real vlogic_high = 5;
52 parameter real vlogic_low = 0;
53 parameter real vtrans_clk = 2.5;
54 parameter real vref = 1.0;
55
56
57 `define NUM_ADC_BITS 10
58 `define MAXINT 2_147_483_647.0
59
60
61 // macro to calculate the fractional mismatches in bits
62 `define FRAC_MM(I) (1.0 + mismatch_fact*(dist_range*abs($random(I
    )/'MAXINT) - \
63     half_dist_range))
64
65     real dist_range, half_dist_range;
66     real comp_var[0:'NUM_ADC_BITS-1]; // Fractional comparator
        mismatches
67     real unconverted;
68     real halfref;
69     real comp_vref;
70     real vd[0:'NUM_ADC_BITS-1];
71     integer i;
72     integer iseed;
73
74     analog begin
75         @ ( initial_step ) begin
76             dist_range = 0.02;
77             half_dist_range = 0.01;
78             generate j ( 0, 'NUM_ADC_BITS-1 ) begin
79                 iseed = j;
80                 comp_var[j] = 'FRAC_MM(iseed);
81             end
82             halfref = vref / 2;
83         end
84
85         @ (cross(V(vclk) - vtrans_clk, 1, 1.0, vclk.potential.
            abstol)) begin
86             unconverted = V(vin);
87             for (i = 'NUM_ADC_BITS-1; i >= 0 ; i = i - 1) begin
88                 vd[i] = 0;
89                 comp_vref = halfref * comp_var[i];

```

```
90         if (unconverted > comp_vref) begin
91             vd[i] = vlogic_high;
92             unconverted = unconverted - comp_vref;
93         end else begin
94             vd[i] = vlogic_low;
95         end
96         unconverted = unconverted * 2;
97     end
98 end
99
100 //
101 // assign the outputs
102 //
103 V(D9) <+ transition( vd[9], tdel, trise, tfall );
104 V(D8) <+ transition( vd[8], tdel, trise, tfall );
105 V(D7) <+ transition( vd[7], tdel, trise, tfall );
106 V(D6) <+ transition( vd[6], tdel, trise, tfall );
107 V(D5) <+ transition( vd[5], tdel, trise, tfall );
108 V(D4) <+ transition( vd[4], tdel, trise, tfall );
109 V(D3) <+ transition( vd[3], tdel, trise, tfall );
110 V(D2) <+ transition( vd[2], tdel, trise, tfall );
111 V(D1) <+ transition( vd[1], tdel, trise, tfall );
112 V(D0) <+ transition( vd[0], tdel, trise, tfall );
113
114 `undef NUM_ADC_BITS
115     end
116 endmodule
```

# APÊNDICE B – Código Verilog-A do DAC ideal de 10 bits

```

1 // VerilogA for digital_analog_conv_block, ideal_dac, veriloga
2
3 'include "constants.vams"
4 'include "disciplines.vams"
5
6 module ideal_dac(d9,d8,d7,d6,d5,d4,d3,d2,d1,d0,vout,vdd,vss,vmin,
7     vmax);
8 parameter real vtrans=0.5;
9 parameter real tdelay = 0;
10 parameter real ttime = 1p;
11
12 inout vdd,vss;
13 input d9,d8,d7,d6,d5,d4,d3,d2,d1,d0;
14 input vmin, vmax;
15 output vout;
16 electrical vout,vdd,vss,d9,d8,d7,d6,d5,d4,d3,d2,d1,d0,vmin,vmax;
17 real result_sum;
18 real result,d_9,d_8,d_7,d_6,d_5,d_4,d_3,d_2,d_1,d_0;
19 analog begin
20     d_9 = V(d9)*512;
21     d_8 = V(d8)*256;
22     d_7 = V(d7)*128;
23     d_6 = V(d6)*64;
24     d_5 = V(d5)*32;
25     d_4 = V(d4)*16;
26     d_3 = V(d3)*8;
27     d_2 = V(d2)*4;
28     d_1 = V(d1)*2;
29     d_0 = V(d0)*1;
30
31     result_sum = d_9 + d_8 + d_7 + d_6 + d_5 + d_4 + d_3 + d_2 +
32         d_1 + d_0;
33
34     result = result_sum * ((V(vmax)-V(vmin))/1024);
35
36     V(vout) <+ transition(result, tdelay, ttime);
37 end
38 endmodule

```

## APÊNDICE C – Código Verilog-A do gerador de rampa

```

1 // VerilogA for comparator_block, offset_ramp_gen, veriloga
2
3 'include "constants.vams"
4 'include "disciplines.vams"
5
6 module offsetrampgenerator(vramp, clk);
7 input clk;
8 output vramp;
9 electrical vramp, clk;
10
11 parameter vth=1; // clk threshold voltage
12 parameter resolution=100u; // resolution of the ramp
13 parameter maxin = 30m; // maximum input level
14 parameter td = 100p; // delay time
15 parameter tr = 100p; // rise time
16
17 real offset;
18 integer direction;
19
20 analog
21 begin
22   @(initial_step("static","tran")) begin // On initialization,
23     offset = maxin-resolution/2;
24     direction = -1 ;
25   end
26
27   @( cross( V(clk)>vth, -1 )) begin
28     offset = offset + (direction * resolution) ;
29   end
30
31   if(offset < -(maxin+(resolution/2))) begin
32     direction= 1 ;
33   end
34
35   V(vramp) <+ transition( offset, td, tr, tr ) ;
36
37 end
38 endmodule

```