



Pedro Miguel Cunha da Silva

Design of A/D converter for a
CMOS-based Magnetic Imaging Sensor

Universidade do Minho
Escola de Engenharia





Universidade do Minho
Escola de Engenharia

Pedro Miguel Cunha da Silva

Design of A/D converter for a
CMOS-based Magnetic Imaging Sensor

Dissertação de Mestrado
Ciclo de Estudos Integrados Conducentes ao Grau de
Mestre em Engenharia Eletrónica Industrial e de Computadores

Trabalho efetuado sob a orientação de
Professor Doutor Luis Alexandre Machado da Rocha
Engenheiro João Paulo Valente de Marques Piteira

DECLARAÇÃO

Nome: Pedro Miguel Cunha da Silva

Endereço eletrónico: a65331@alumni.uminho.pt Telefone: 914345829

Bilhete de Identidade/Cartão do Cidadão: 14322803

Título da dissertação: Design of A/D converter for a CMOS-based Magnetic Imaging Sensor

Orientador:

Luis Alexandre Machado da Rocha

Ano de conclusão: 2016

Mestrado em Engenharia Eletrónica Industrial e de Computadores

É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE.

Universidade do Minho, ____/____/____

Assinatura:

AGRADECIMENTOS

Apesar de todos os documentos homólogos a este ficarem lembrados como sendo o trabalho de apenas uma pessoa, na verdade a grande maioria deles, para não dizer a sua totalidade, representa o esforço, dedicação e conhecimento de várias entidades. Por isso, a minha dissertação não é exceção. Desde logo um merecido agradecimento da minha parte ao professor Luis Rocha que, quando abordado por mim, se mostrou disponível para ser meu orientador e que ao longo de todo o processo sempre se mostrou disponível para me ajudar em tudo aquilo que necessitasse.

Também à INL e a todos os seus colaboradores por mim abordados gostaria de deixar uma mensagem de apreço. Ao João Piteira que desde início mostrou entusiasmo e confiança em mim para realizar este projeto, ao Álvaro Geraldes e Filipe Alves que me auxiliaram com os seus conhecimentos no *software Cadence* quando me deparei com dificuldades e ao Miguel Guedes, que se revelou uma ajuda preciosa na vertente digital deste projeto.

Aos meus colegas Renato Castro, Fábio Martins, Fábio Leitão, Carlos Silva e Carlos Ferreira fica também aqui um forte agradecimento pois para além de me darem sugestões e conselhos no decorrer de conversas na universidade, também me brindaram com momentos de boa disposição ao longo de toda esta jornada.

Por último, e com certeza mais importante, fica aqui um enorme agradecimento a toda a minha família, em particular à minha mãe que sempre me ajudou e incentivou quando a vida nos pregou algumas rasteiras e ao meu pai que, onde quer que esteja, sempre me ajudou e continuará a ajudar na concretização dos sonhos que ainda tenho por realizar.

A todas estas entidades, um enorme bem-haja.

RESUMO

Este trabalho tem como principal objetivo o desenvolvimento de um Conversor Analógico Digital (*ADC*), desde a definição da arquitetura, simulação de alto e baixo nível e implementação em *CMOS*.

O *ADC* a ser desenvolvido faz parte de um sistema que tem como objetivo principal o mapeamento magnético de diferentes corpos através de diversos sensores analógicos concebidos para o efeito, de maneira a que essa mesma imagem magnética possa ser posteriormente processada e analisada num sistema computadorizado. Devido à natureza analógica da informação proveniente dos sensores magnéticos, existe a necessidade de fazer a conversão das tensões provenientes destes para formato digital, para que todo o processo de análise desta informação possa ser uma realidade.

Como tal, a inclusão de um *ADC* em tal sistema é algo imprescindível para que o objetivo proposto possa ser atingido com êxito. Deste modo, de maneira a que um elemento tão fulcral, como é um *ADC* em sistemas deste género, não seja um elemento perturbador ao bom funcionamento de todo o processo de aquisição de informação, um determinado número de restrições (e.g. consumo energético e área de silício ocupada) têm que ser sempre tidas em atenção pois qualquer desrespeito grosseiro para com qualquer uma delas pode revelar-se catastrófico. Sendo este *ADC* incluído num sistema com tais particularidades, a aplicação necessita de uma taxa de amostragem de 3 *MSPS* e uma resolução de 12 *bits* para que este dispositivo possa vir a desempenhar as funções para as quais está a ser concebido (e.g. mapeamento magnético de células presentes numa lamela de microscópio).

Neste sentido, a arquitetura *SAR* foi a escolhida, sendo a tecnologia a *CMOS AMS 0.35 μm* . No final, foi concebido e simulado em pós-layout um *ADC* capaz de realizar conversões a uma taxa de 3 *MSPS* quando alimentado a 3.3 V. A área ativa final que o *ADC* apresenta é de 0.2625 mm^2 .

Palavras-Chave: Conversor Analógico Digital, Simulação, Implementação e Dimensionamento

ABSTRACT

The main goal of this work is the development of an analogic to digital converter (*ADC*), since the definition of its architecture, low level simulation and *CMOS* implementation.

This *ADC* will be part of a major system whose purpose is the magnetic mapping of different objects, helped by several analogic sensors designed for this objective. As the nature of this sensors is analogic, some entity as the responsibility to convert this information for a digital representation, in order to process the data in a computerized system.

Because of this necessity, the inclusion of an *ADC* is essential for the main goal to be reached. Thus, in order to prevent any trouble for the final system, this *ADC* has to respect some restrictions (e.g. power consumption and silicon area) because any disrespect with any one of them could reveal catastrophic consequences. Regarding the particularities of the final system, this *ADC* should be able to accomplish conversions at a rate of 3 *MSPS* (Mega Samples per Second) with 12 bits of resolution.

In this way, a *SAR* architecture was chosen and simulated in *CMOS AMS 0.35 μm* technology. At the end, was designed and simulated in post layout an *ADC* capable to convert at a rate of 3 *MSPS* when supplied by 3.3 V. The final active area occupied by this *ADC* is 0.2625 mm^2 .

KEYWORDS: ANALOGIC DIGITAL CONVERTER, SIMULATION, IMPLEMENTATION AND DESIGN.

ÍNDICE

| | |
|--|------|
| Agradecimentos | iii |
| Resumo | v |
| Abstract..... | vii |
| Índice | ix |
| Lista de Figuras | xiii |
| Lista de Tabelas | xv |
| Lista de Abreviaturas e acrónimos | xvii |
| 1. Introdução..... | 1 |
| 1.1 Motivação e Objetivos | 2 |
| 1.2 Especificações..... | 4 |
| 1.3 Estrutura da dissertação | 4 |
| 2. Estudo de Arquiteturas de Conversores | 5 |
| 2.1 Arquiteturas de Conversores..... | 5 |
| 2.2 Comparação de Arquiteturas | 6 |
| 2.3 Escolha da arquitetura..... | 6 |
| 2.4 Conversor <i>SAR</i> | 7 |
| 2.4.1 Conceito <i>SAR ADC</i> | 7 |
| 2.4.2 <i>Switched-Capacitor SAR ADC</i> | 7 |
| 2.4.3 Topologia <i>Fully-Differential vs. Single-ended</i> | 8 |
| 2.4.4 Princípio de funcionamento..... | 9 |
| 2.4.5 Exemplo de conversão..... | 9 |
| 3. Simulação de alto nível | 17 |
| 3.1 Banco de condensadores ou <i>DAC</i> | 17 |
| 3.1.1 <i>Split Capacitor arrays</i> | 18 |
| 3.1.2 Dimensionamento dos condensadores..... | 19 |
| 3.1.2.1 Condensador unitário | 19 |
| 3.1.2.2 Ruído <i>kT/C</i> | 19 |
| 3.1.2.3 Estudo de tolerâncias..... | 20 |
| 3.1.2.4 Condensadores em série | 27 |

| | | |
|---------|---|----|
| 3.1.3 | Representação em <i>Matlab</i> | 27 |
| 3.2 | Máquina de estados | 28 |
| 3.3 | Vista geral..... | 29 |
| 3.3.1 | Análise de <i>INL</i> e <i>DNL</i> | 30 |
| 4. | Simulação de baixo nível..... | 31 |
| 4.1 | Comparador | 31 |
| 4.1.1 | Modo de auto-calibração..... | 32 |
| 4.1.1.1 | Princípio de funcionamento | 33 |
| 4.1.2 | Tempo de estabilização do <i>output</i> | 33 |
| 4.1.3 | Análise de <i>offset</i> | 35 |
| 4.2 | Amplificador operacional..... | 36 |
| 4.2.1 | Variação da tensão de entrada..... | 38 |
| 4.2.2 | Análise em frequência..... | 39 |
| 4.2.2.1 | Análise em malha aberta..... | 39 |
| 4.2.2.2 | Análise em malha fechada | 40 |
| 4.3 | Banco de condensadores ou <i>DAC</i> | 41 |
| 4.3.1 | Dimensionamento dos condensadores | 43 |
| 4.3.2 | Dimensionamento dos comutadores | 43 |
| 4.3.3 | Análise de frequências de entrada..... | 45 |
| 4.4 | Máquina de estados | 47 |
| 4.4.1 | Conceção do circuito..... | 48 |
| 4.4.2 | Comportamento da máquina de estados..... | 48 |
| 4.5 | Circuito final..... | 50 |
| 5. | Fase de <i>Layout</i> | 53 |
| 5.1 | Comparador | 53 |
| 5.2 | Amplificador operacional..... | 55 |
| 5.3 | Banco de condensadores ou <i>DAC</i> | 56 |
| 5.4 | Máquina de estados | 56 |
| 5.5 | Circuito final..... | 58 |
| 6. | Simulação pós- <i>Layout</i> | 59 |

| | | |
|-------|---|----|
| 6.1 | Comparador | 59 |
| 6.1.1 | Tempo de estabilização do <i>output</i> | 59 |
| 6.1.2 | Análise de <i>offset</i> | 60 |
| 6.2 | Amplificador operacional | 61 |
| 6.3 | Banco de condensadores ou <i>DAC</i> | 62 |
| 6.4 | Máquina de estados | 63 |
| 6.5 | Circuito final | 64 |
| 7. | Conclusões | 65 |
| 7.1 | Trabalho futuro | 66 |
| 7.2 | Melhorias | 66 |
| | Bibliografia | 67 |
| | Anexo I – Comparador | 69 |
| | Anexo II – <i>Process Corners</i> | 71 |
| | Anexo III – Amplificador operacional | 72 |
| | Anexo IV – Banco de Condensadores ou <i>DAC</i> | 74 |
| | Anexo V – Máquina de estados | 75 |

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1.1 Vista geral do chip em desenvolvimento | 2 |
| Figura 2.1 Arquiteturas de conversores, aplicações, resoluções e taxas de amostragem (adaptado)..... | 5 |
| Figura 2.2 Vista geral de um <i>SAR ADC</i> | 7 |
| Figura 2.3 Vista geral de um <i>Switched-Capacitor SAR ADC</i> | 8 |
| Figura 2.4 Fase de <i>Sample</i> | 10 |
| Figura 2.5 Fase de <i>Sample</i> (circuito equivalente)..... | 10 |
| Figura 2.6 Fase de <i>Hold</i> | 11 |
| Figura 2.7 Fase de <i>Hold</i> (circuito equivalente) | 11 |
| Figura 2.8 Comparação 11° <i>bit</i> | 12 |
| Figura 2.9 Comparação 11° <i>bit</i> (circuito equivalente)..... | 13 |
| Figura 2.10 Comparação 10° <i>bit</i> | 13 |
| Figura 2.11 Comparação 10° <i>bit</i> (circuito equivalente)..... | 14 |
| Figura 2.12 Comparação 10° <i>bit</i> com 11° <i>bit</i> com valor 0..... | 15 |
| Figura 2.13 Comparação 10° <i>bit</i> com 11° <i>bit</i> com valor 0 (circuito equivalente) | 15 |
| Figura 3.1 dois <i>arrays</i> balanceados com um condensador em série ($C_{total} \approx 96 * C$) | 18 |
| Figura 3.2 três <i>arrays</i> balanceados com dois condensadores em série ($C_{total} \approx 40 * C$) | 18 |
| Figura 3.3 quatro <i>arrays</i> balanceados com três condensadores em série ($C_{total} \approx 28 * C$)..... | 18 |
| Figura 3.4 C unitário em função de V ruído | 20 |
| Figura 3.5 <i>DAC</i> com três <i>arrays</i> balanceados, na primeira fase de conversão | 21 |
| Figura 3.6 <i>DAC</i> com três <i>arrays</i> balanceados, na primeira fase de conversão (primeiro circuito equivalente) | 22 |
| Figura 3.7 <i>DAC</i> com três <i>arrays</i> balanceados, na primeira fase de conversão (segundo circuito equivalente) | 22 |
| Figura 3.8 <i>DAC</i> com três <i>arrays</i> balanceados, na primeira fase de conversão (circuito equivalente final) | 23 |
| Figura 3.9 Distribuição de V test em <i>DAC</i> com três <i>arrays</i> balanceados..... | 24 |
| Figura 3.10 <i>DAC</i> com quatro <i>arrays</i> balanceados, na primeira fase de conversão | 25 |
| Figura 3.11 <i>DAC</i> com quatro <i>arrays</i> balanceados, na primeira fase de conversão (circuito equivalente final) | 25 |

| | |
|---|----|
| Figura 3.12 Distribuição de V test em <i>DAC</i> com quatro <i>arrays</i> balanceados..... | 26 |
| Figura 3.13 Representação do <i>DAC</i> em <i>Simulink</i> | 27 |
| Figura 3.14 Representação da máquina de estados em <i>Simulink</i> | 28 |
| Figura 3.15 Representação do <i>ADC</i> em <i>Simulink</i> | 29 |
| Figura 3.16 Valores de <i>DNL</i> e <i>INL</i> simulados | 30 |
| Figura 4.1 Comparador com circuito de auto-calibração (adaptado)..... | 32 |
| Figura 4.2 Tempo de estabilização da saída do comparador | 34 |
| Figura 4.3 Configuração do comparador para estudo do <i>offset</i> de entrada..... | 35 |
| Figura 4.4 <i>Offsets</i> obtidos depois da calibração..... | 36 |
| Figura 4.5 Amplificador operacional (adaptado)..... | 37 |
| Figura 4.6 Amplificador configurado como seguidor de tensão..... | 37 |
| Figura 4.7 Diferença entre tensão ideal e saída do amplificador | 38 |
| Figura 4.8 Diagrama de Bode do amplificador em malha aberta | 39 |
| Figura 4.9 Diagrama de Bode para amplificador configurado como seguidor de tensão..... | 40 |
| Figura 4.10 Configuração do <i>DAC</i> | 42 |
| Figura 4.11 Configuração dos comutadores em baixo nível..... | 45 |
| Figura 4.12 <i>FFT</i> de saída e ideal com frequência de entrada de 164 <i>kHz</i> | 46 |
| Figura 4.13 <i>FFT</i> de saída e ideal com frequência de entrada de 726 <i>kHz</i> | 46 |
| Figura 4.14 <i>FFT</i> de saída e ideal com frequência de entrada de 1.43 <i>MHz</i> | 46 |
| Figura 4.15 Interface da máquina de estados | 50 |
| Figura 4.16 Sinais durante uma conversão | 51 |
| Figura 5.1 <i>Layout</i> do comparador | 54 |
| Figura 5.2 <i>Layout</i> do amplificador operacional | 55 |
| Figura 5.3 <i>Layout</i> do banco de condensadores | 56 |
| Figura 5.4 <i>Layout</i> da máquina de estados | 58 |
| Figura 5.5 <i>Layout</i> do circuito final..... | 58 |
| Figura 6.1 Tempo de estabilização do <i>output</i> do comparador (<i>pós-layout</i>)..... | 60 |
| Figura 6.2 <i>Offset</i> de entrada do comparador | 61 |
| Figura 6.3 <i>Offset</i> de saída do amplificador operacional (<i>pós-layout</i>)..... | 62 |
| Figura 6.4 <i>FFT</i> de saída e ideal com frequência de entrada de 164 <i>kHz</i> (<i>pós-layout</i>)..... | 62 |
| Figura 6.5 <i>FFT</i> de saída e ideal com frequência de entrada de 726 <i>kHz</i> (<i>pós-layout</i>)..... | 63 |
| Figura 6.6 <i>FFT</i> de saída e ideal com frequência de entrada de 1.43 <i>MHz</i> (<i>pós-layout</i>) | 63 |
| Figura 6.7 Sinais durante uma simulação (<i>pós-layout</i>)..... | 64 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 Comparação entre atributos de diferentes conversores (adaptado) | 6 |
| Tabela 2 Dimensões dos <i>mosfets</i> que constituem o comparador..... | 69 |
| Tabela 3 Capacidade e dimensões dos condensadores que constituem o comparador | 70 |
| Tabela 4 Dimensões dos <i>mosfets</i> que constituem o amplificador operacional..... | 72 |
| Tabela 5 Capacidade e dimensão do condensador C_c presente no amplificador operacional . | 72 |
| Tabela 6 Resistências e dimensões das resistências que constituem o amplificador operacional | 72 |
| Tabela 7 Dimensões e capacidade/resistência dos elementos adicionais da topologia em seguidor de tensão | 73 |
| Tabela 8 Dimensões dos <i>mosfets</i> que constituem os comutadores presentes no <i>DAC</i> | 74 |

LISTA DE ABREVIATURAS E ACRÓNIMOS

| | |
|---------------|--|
| ADC | Analogic to Digital Converter |
| CMOS | Complementary Metal-Oxide-Semiconductor |
| DAC | Digital to Analog Converter |
| dB | Decibel |
| DNL | Differential Nonlinearity |
| F | Faraday |
| FFT | Fast Fourier Transform |
| ENOB | Effective Number Of Bits |
| INL | Integral Nonlinearity |
| K | Graus Kelvin |
| MOSFET | Metal Oxide Semiconductor Field Effect Transistor |
| MSPS | Mega Samples Per Second |
| s | Segundos |
| SAR | Successive Approximation Register |
| SFDR | Spurious-free dynamic range |
| SNDR | Signal to Noise and Distortion Ratio |
| V | Volts |
| °C | Graus Celcius |
| Ω | Ohm |

1. INTRODUÇÃO

Em toda a história da Humanidade, o magnetismo tem assumido um papel preponderante nas nossas vidas. Desde a época dos descobrimentos, onde instrumentos como uma simples bússola revolucionaram a forma como os nossos antepassados navegavam e se orientavam nos mares, até aos dias de hoje, onde o fenómeno do magnetismo é um dos principais impulsionadores para o crescimento tecnológico astronómico que temos vindo a assistir nas últimas décadas.

Grandes nomes da nossa história como Michael Faraday, que descobriu aquilo a que hoje chamamos de indução magnética, James Clerk Maxwell, que sintetizou a grande descoberta de Faraday sob a forma das famosas equações de Maxwell, ou Nikola Tesla, o inventor do motor de indução e do sistema trifásico de energia elétrica que chega às casas de todos nós, são ainda hoje lembrados não apenas porque os seus nomes representam grandezas físicas, mas sobretudo porque as suas descobertas foram imprescindíveis para o avanço a que a civilização como a conhecemos hoje tem vindo a ser sujeita.

No nosso dia-a-dia, e muitas vezes sem nos apercebermos disso, utensílios como um simples aspirador para nos ajudar nas tarefas domésticas ou um motosserra elétrico para nos auxiliar na preparação da lenha para o inverno, têm por detrás do seu funcionamento o princípio do magnetismo. Mas não é só nas tarefas quotidianas que este princípio nos ajuda. Na indústria em geral, e utilizando a indústria médica a título de exemplo, esta grandeza física assume um papel bastante importante, ora não se chamasse um dos mais fiáveis exames médicos de ressonância magnética.

Associada a toda esta tecnologia, claro está que o desenvolvimento de sensores que fossem dotados da capacidade de “ler” intensidades de campos magnéticos tornou-se uma necessidade. Não só no caso da ressonância magnética como em tecnologias usadas noutras indústrias, é necessário o uso de alguma entidade que seja responsável pela conversão dessa grandeza física, o campo magnético, numa grandeza elétrica pois esta última pode ser mais facilmente processada.

Apesar das vantagens anteriormente mencionadas, existe ainda uma barreira entre a informação conseguida pelos sensores e o seu posterior processamento que necessita ser ultrapassada. Isto porque a informação proveniente destes sensores é de natureza analógica e o processamento de informação é feito digitalmente. É nesta fase que o tema principal desta

dissertação é abordado, sendo a sua motivação e objetivos principais enumerados no tópico seguinte.

1.1 Motivação e Objetivos

No mundo em que vivemos atualmente, cada vez mais informatizado e onde a natureza da esmagadora maioria da informação que é processada diariamente é de natureza digital, torna-se evidente que, de algum modo, alguma entidade é responsável pela conversão da informação de natureza analógica para digital, uma vez que, contrariamente à informação processada, a grande maioria da informação captada no meio que nos rodeia é de natureza analógica. É essencialmente na implementação de uma entidade deste género em que esta dissertação se debruça, abordando todos os passos inerentes à sua conceção desde a definição da arquitetura, simulação de alto e baixo nível e implementação em *CMOS*.

O conversor analógico digital a ser desenvolvido será parte integrante de um *chip* já em desenvolvimento que tem como objetivo nuclear o de fazer mapeamento magnético de diferentes corpos através de diversos sensores analógicos concebidos para o efeito, de maneira a que essa mesma imagem magnética possa ser posteriormente processada e analisada num sistema computadorizado. A figura 1.1 mostra aquele que é o conceito do sistema que terá como um dos módulos integrantes o conversor a ser desenvolvido nesta dissertação.

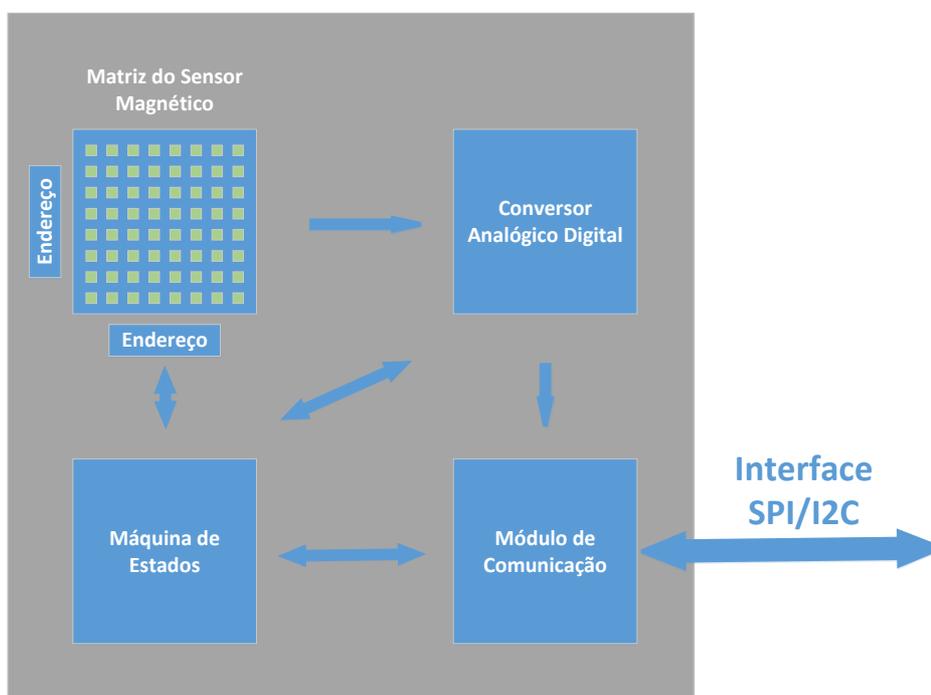


Figura 1.1 Vista geral do chip em desenvolvimento

Apesar de no mercado existirem inúmeras propostas de conversores dotados da capacidade de fazer este tipo de tarefa, a inclusão de tais entidades no sistema em desenvolvimento é algo que não poderá ser tido em mente uma vez que um dos principais objetivos deste sistema é a integração de um *array* de sensores magnéticos, da eletrónica de leitura e da conversão da informação proveniente desse *array* de sensores num único *chip*. Deste modo torna-se necessária a implementação de um conversor analógico digital nesse mesmo *chip*.

Devido ao facto de existirem várias arquiteturas (consultar capítulo 2 – Estudo de arquiteturas de conversores) que permitem este tipo de conversão, é necessário analisar cada uma delas e retirar quais os seus pontos mais ou menos positivos de maneira a que a escolha da arquitetura recaia sobre aquela que apresenta um melhor balanço entre todos os vários parâmetros, tendo em conta aqueles que são os requisitos da aplicação destino.

No *cômputo* geral, o estudo e implementação deste conversor é importante pois a inclusão de tal bloco é essencial para que o processamento da informação proveniente dos sensores magnéticos por parte de sistemas computadorizados possa ser possível. Outro aspeto importante é também o facto do nível de integração ser bastante alto, ideia essa já patente desde a idealização deste *chip*.

Deste modo, o foco principal desta dissertação é o de conceber um conversor analógico digital, o que faz com que o item a encabeçar a lista de objetivos seja, sem margem para dúvidas, o de conseguir desenvolver um bloco que permita fazer conversões de tensões analógicas para digitais respeitando todos os requisitos que a aplicação final exige.

Para que tal seja possível, e tendo em mente o cariz sequencial da metodologia a ser aplicada neste trabalho, é estritamente necessário que objetivos intermédios sejam progressivamente cumpridos com sucesso. Desta forma, todo o processo de concessão deste sistema torna-se bastante mais sólido e consistente.

De entre todos os objetivos secundários existentes, os mais preponderantes estão mencionados a seguir:

- Escolher a arquitetura do conversor;
- Simular a arquitetura com tolerâncias do processo de fabrico;
- Conceber e simular os circuitos eletrónicos que serão parte integrante do Conversor;
- Desenhar o *layout* do conversor.

1.2 Especificações

Estando nesta fase já bem idealizado e planeado qual o papel a desempenhar pelo *chip* a desenvolver e qual a sua aplicação final, chegou-se à conclusão que, no que ao *ADC* diz respeito, este teria de ser dotado de 12 *bits* de resolução com uma taxa de amostragem de 3 *MSPS*.

Relativamente à tensão de alimentação, esta terá o valor de 3.3 V, sendo todo o chip realizado e pensado na tecnologia *CMOS AMS 0.35 μm*.

1.3 Estrutura da dissertação

O primeiro capítulo desta dissertação apresenta por traços muito gerais qual o tema que é tratado, tendo por isso apenas um carácter introdutório. Relativamente ao capítulo seguinte, ele pretende mostrar quais são as arquiteturas de conversores mais difundidas no mercado bem como as características de cada uma delas. Neste sentido, é também nesta seção que é definida a arquitetura mais conveniente para o sistema a desenvolver. Posto isto, no capítulo três é feito um estudo em alto nível da arquitetura escolhida, recorrendo para tal ao *software Matlab*. De seguida, o capítulo referente à simulação de baixo nível toma lugar. É no decorrer desta seção que são analisados e simulados todos os diferentes blocos que constituem o *ADC* final, procedendo deste modo ao dimensionamento de cada um dos elementos que os constituem. Concluída esta etapa, é mostrado o *layout* resultante de todos os diferentes blocos constituintes do *ADC* bem como o *layout* final de todo o sistema. No capítulo seguinte são mostrados os resultados obtidos das simulações de pós-*layout* realizadas não só ao sistema final como também a todos os blocos que o constituem. Por último, são feitas as considerações finais desta dissertação abordando o trabalho a realizar futuramente bem como melhorias de passível inclusão no sistema final.

2. ESTUDO DE ARQUITETURAS DE CONVERSORES

São inúmeras as arquiteturas que permitem fazer a conversão de tensões analógicas para digitais.

Deste modo, existindo então diferentes escolhas neste género de sistemas, torna-se evidente que cada uma delas apresenta os seus pontos mais ou menos fortes que, dependendo dessas mesmas características, melhor se adaptam às diferentes situações e contextos de funcionamento.

2.1 Arquiteturas de Conversores

De entre todas as arquiteturas existentes, as mais difundidas são sem dúvida as *Sigma-Delta* [1], *Pipeline* [2] e *SAR* [3] devido aos seus princípios de funcionamento serem relativamente intuitivos e apresentarem um equilíbrio entre as suas características bastante bom. De maneira a melhor compreender qual a gama de taxas de conversão, bem como as resoluções das diferentes arquiteturas, a figura 2.1, adaptada de [4], elucida onde é que estas se situam.

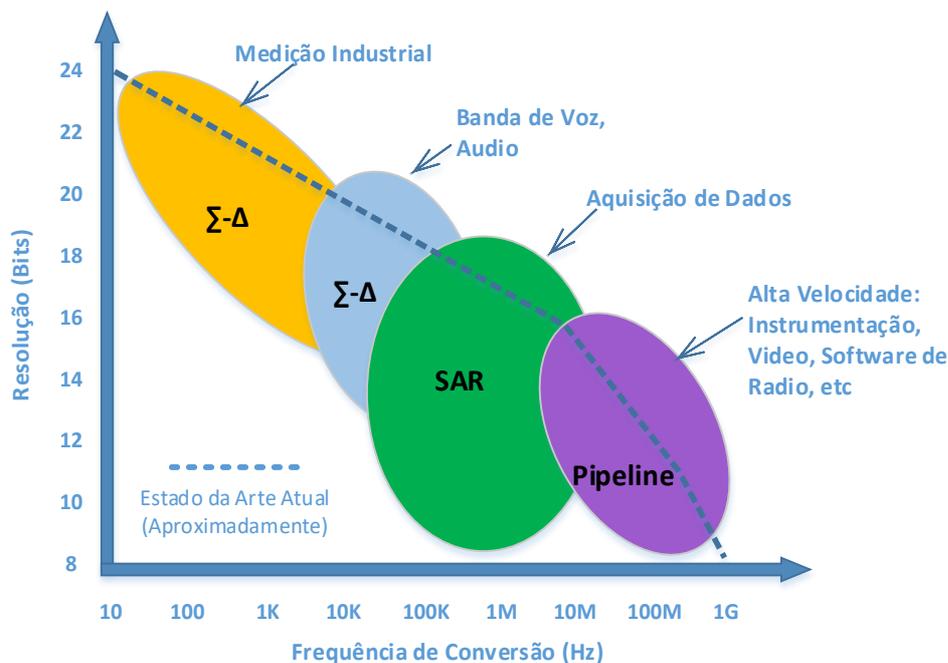


Figura 2.1 Arquiteturas de conversores, aplicações, resoluções e taxas de amostragem (adaptado)

2.2 Comparação de Arquiteturas

Apesar dos princípios de funcionamento destas topologias de conversores serem relativamente simples, as suas características em termos de funcionamento são bastante dispare, o que se reflete em atributos de baixo nível (e.g. área de silício, consumo energético, complexidade de implementação) também eles bastante diferentes.

A tabela 1, adaptada de [5], classifica cada uma das arquiteturas aqui abordadas quanto aos vários atributos que normalmente são avaliados aquando da apreciação deste tipo de sistemas.

Tabela 1 Comparação entre atributos de diferentes conversores (adaptado)

| Architecture | Latency | Speed | Resolution | Area and power consumption |
|---------------------|----------------|--------------|-------------------|-----------------------------------|
| Sigma-delta | High | Slow-medium | High | Medium |
| Pipeline | Low-medium | High | Medium | Medium |
| SAR | Low | High | Medium | Low |

2.3 Escolha da arquitetura

Depois de recolhida e analisada a informação apresentada sobre conversores, fazendo um balanço das principais características que este *ADC* deve apresentar (3 *MSPS*; 12 *bits*), torna-se evidente que a arquitetura mais adequada a adotar é a *SAR*, ou seja, conversão por aproximações sucessivas. Fazendo uma análise mais detalhada da figura 2.1, é visível que esta arquitetura em particular é mais do que ajustada para as especificações impostas, tendo ainda a vantagem de apresentar, comparativamente às demais arquiteturas, baixa latência, baixa área ocupada e também baixo consumo energético (tabela 1).

É importante referir que existem mais arquiteturas para além das três aqui mencionadas mas, sendo estas as mais enraizadas e com mais provas dadas no mercado atual, tornam-se naquelas que mais informação apresentam relativamente ao seu funcionamento. Deste modo, o início deste capítulo recai sobretudo sobre a comparação destas três arquiteturas em particular.

2.4 Conversor SAR

Depois de consultada e avaliada uma grande variedade de informação especializada, a arquitetura a ser adotada neste *ADC* acabou por ser a conversão por aproximações sucessivas, ou *SAR*. Deste modo, ao longo desta secção pretende-se mostrar com maior detalhe quais são os elementos constituintes deste *ADC* bem como o seu princípio de funcionamento.

2.4.1 Conceito SAR ADC

Esta arquitetura em particular, tal como todas as outras, tem como objetivo a conversão de uma tensão analógica num valor digital mas, ao contrário das demais, o seu funcionamento é bastante peculiar. Ela tem como blocos fundamentais um comparador, um andar de *Sample & Hold*, um conversor digital-analógico (*DAC*) e uma máquina de estados responsável por toda a dinâmica que o processo de conversão exige [6]. A figura 2.2 pretende representar aquela que é a topologia típica de um *ADC* deste tipo.

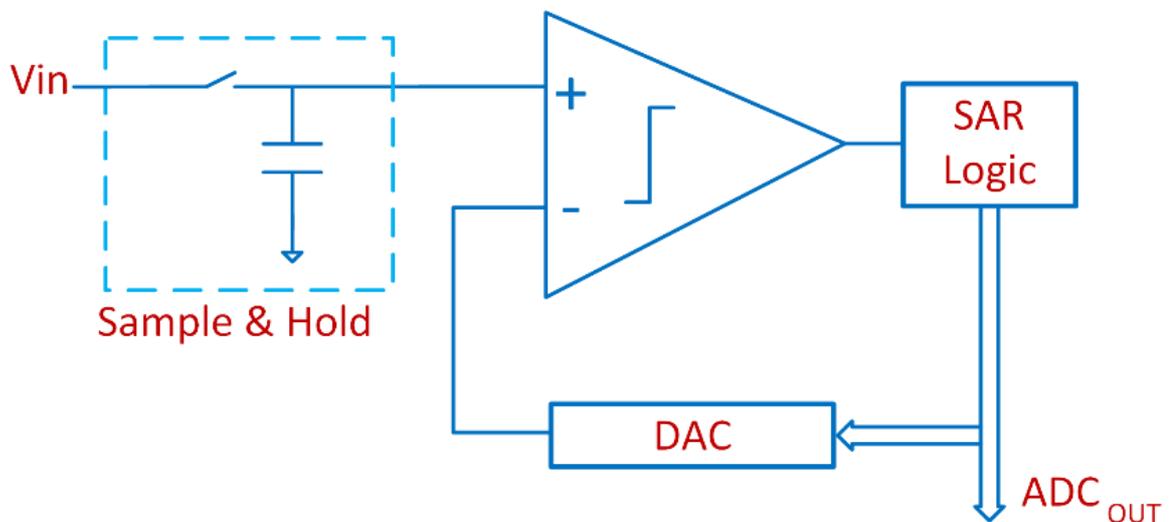


Figura 2.2 Vista geral de um SAR ADC

2.4.2 Switched-Capacitor SAR ADC

Apesar de cada um dos blocos representados na figura 2.2 ser indispensável para um bom funcionamento do *ADC*, é possível combinar dois deles num só. Esses dois blocos são o *DAC* e o andar de *Sample & Hold*. Desta forma, o *ADC* pode ser classificado como *Switched-*

Capacitor SAR ADC (ou *Charge-Redistribution SAR ADC*), que é o esquema mais utilizado na atualidade neste tipo de ADC e também o adotado no decorrer desta dissertação. A figura 2.3 pretende ilustrar o conceito de um *Switched-Capacitor SAR ADC*.

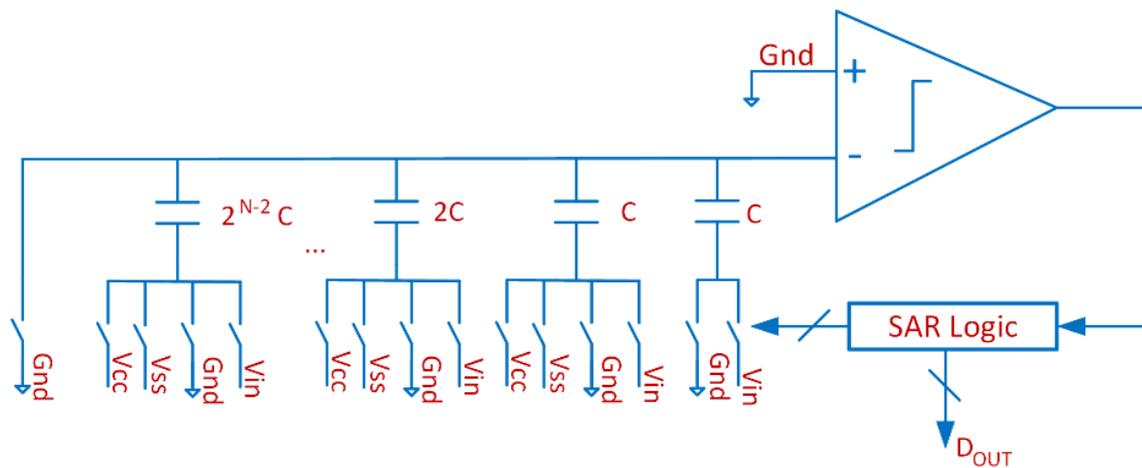


Figura 2.3 Vista geral de um *Switched-Capacitor SAR ADC*

Apesar de ambos os blocos *DAC* e *Sample & Hold* representarem entidades diferentes, e por forma a simplificar a compreensão de qualquer explicação feita adiante, o bloco que engloba tanto o *DAC* como o *Sample & Hold* será apenas denominado de *DAC*.

2.4.3 Topologia *Fully-Differential vs. Single-ended*

Ainda dentro da arquitetura *SAR*, existe a possibilidade de ter ora uma entrada com topologia *fully-differential* ou *single-ended*. No que ao desempenho de todo o sistema diz respeito, uma topologia *fully-differential* apresenta um quadro bastante mais vantajoso pois apresenta imunidade a ruído de modo comum, cancela os harmónicos provenientes das não linearidades do circuito e aumenta também a gama dinâmica de entrada para o dobro, relaxando assim requisitos impostos ao comparador [6]. No entanto, esta topologia requer uma área de silício mais elevada, o que resulta em potências consumidas também elas mais elevadas e em custos de fabrico mais dispendiosos.

Devido aos fatores anteriormente mencionados, no decorrer desta dissertação será usada uma topologia *single-ended*.

2.4.4 Princípio de funcionamento

Como já referido no decorrer desta dissertação, o objetivo de todos os conversores é fazer a representação digital de uma tensão analógica. No entanto, dependendo da sua topologia, o princípio de funcionamento é diferente de todos os outros.

Estando nesta fase já todos os elementos que constituem um *Switched-Capacitor SAR ADC* clarificados, o próximo passo será então perceber como é que todos eles trabalham em conjunto de modo a conseguir fazer uma conversão.

Numa primeira fase, é necessário que a tensão analógica a ser convertida seja armazenada. A entidade responsável por tal tarefa é o *DAC*, que, comandado pela máquina de estados, tem todos os comutadores que o envolvem rearranjados para que a tensão de entrada seja “guardada”. Nas fases posteriores, que são pelo menos tantas quantas o número de *bits* de resolução do *ADC*, os comutadores vão sendo novamente rearranjados á medida que o processo de conversão se vai adiantando, providenciando assim ao comparador todas as tensões de comparação necessárias.

Depois da conversão do último *bit*, e dependendo do *ADC* em questão, o processo de conversão pode voltar ao início ou poderão ser necessárias mais algumas fases para calibração de alguns elementos constituintes do *ADC*.

2.4.5 Exemplo de conversão

Depois de explicada qual a lógica e a dinâmica associada à conversão de uma tensão analógica em digital de um *SAR ADC*, nada melhor do que um exemplo de uma conversão para que todo o conceito fique bem claro. Para uma melhor compreensão da explicação que se segue, assume-se 12 como sendo o número de *bits* do *ADC*, N , uma tensão a converter (V_{in}) com um valor de 3.3 V e as tensões de V_{cc} e V_{ss} como 3.3 V e -3.3 V, respetivamente.

Para começar, a tensão a converter (V_{in}) é armazenada nos condensadores presentes no *DAC*. A figura 2.4 assinala quais os comutadores do *DAC* que se encontram em curto-circuito nesta fase da conversão.

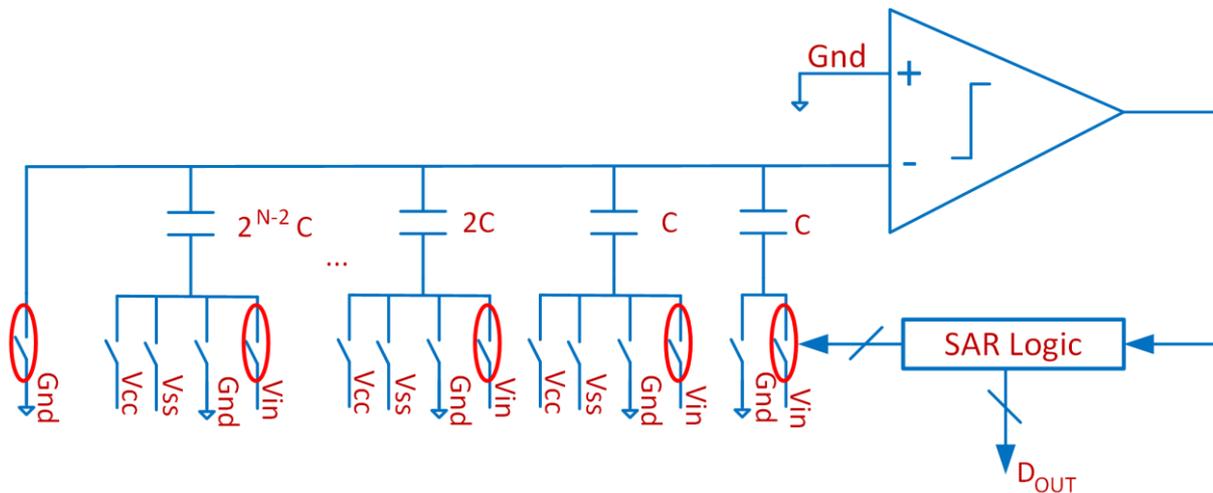


Figura 2.4 Fase de Sample

Com uma análise mais detalhada da figura 2.4, é de rápida perceção que existe um condensador em particular que apenas se encontra ligado a dois comutadores em vez de quatro. Este condensador adicional com capacidade C é necessário para que seja possível uma divisão exata por dois da tensão de referência numa fase posterior da conversão uma vez que a capacidade total fica com o valor de $2^{N-1} * C$ em vez de $(2^{N-1} - 1) * C$ [6].

Outro aspeto que se destaca nesta fase é o facto de que todos os condensadores se encontram em paralelo. Deste modo, e lembrando que neste exemplo o número de *bits* é 12, o circuito equivalente nesta fase é mostrado na figura 2.5.

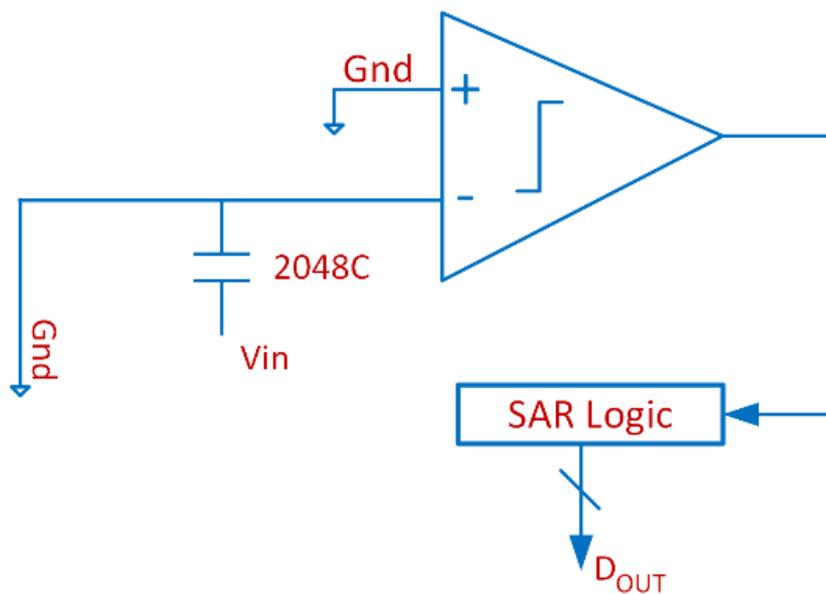


Figura 2.5 Fase de Sample (circuito equivalente)

Nesta altura o “condensador” presente no circuito equivalente (figura 3.4) fica com a carga correspondente à tensão que se pretende converter (3.3 V). Esta fase de conversão é denominada de *Sample*.

Depois de armazenada a tensão a converter, segue-se a fase de *Hold*. A figura 2.6 assinala qual o estado dos comutadores nesta fase.

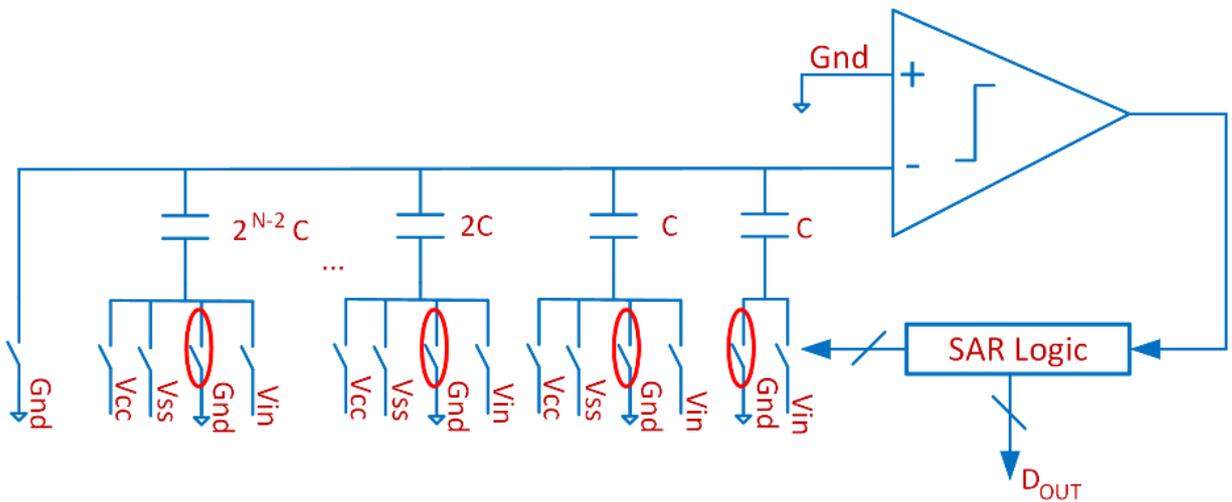


Figura 2.6 Fase de Hold

Tendo em conta a figura anterior, o seu circuito equivalente é mostrado na figura 2.7.

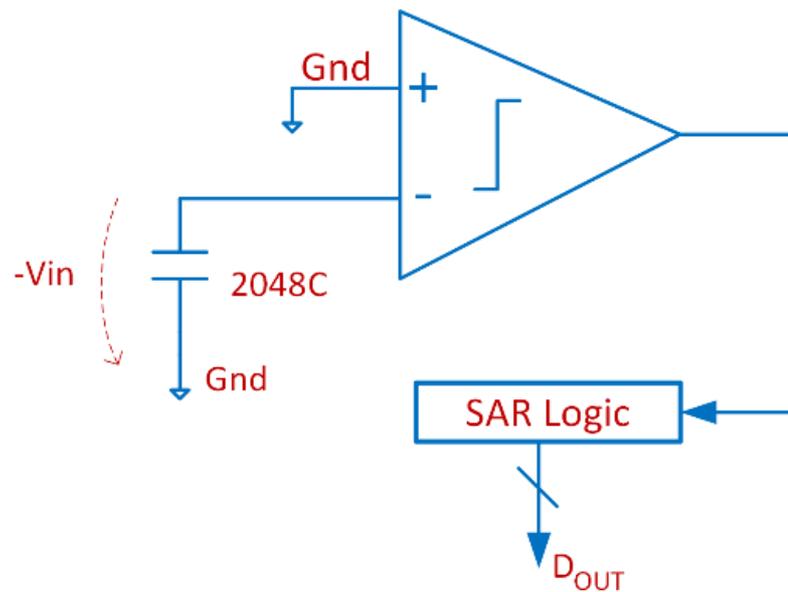


Figura 2.7 Fase de Hold (circuito equivalente)

Visto que a extremidade inferior do condensador está desta feita ligada a Gnd , o potencial que se encontra presente na ligação entre o comparador e o condensador é agora de $-Vin$ (-3.3 V).

Posteriormente a esta fase, e não alterando nada no circuito, é executada a primeira comparação, originando assim o *bit* mais significativo da palavra de 12 *bits*. Este *bit*, para além do mais significativo, também tem um papel fundamental no processo de conversão pois é ele que dita o uso ora dos comutadores conectados a Vcc ou Vss .

Neste exemplo em particular, o *bit* mais significativo será 1 pois a entrada negativa do comparador tem um valor de -3.3 V. Desta forma, a máquina de estados irá manipular os comutadores conectados a Vcc . Caso este *bit* tivesse o valor 0, os comutadores conectados a Vss seriam os utilizados.

Dando seguimento ao processo, segue-se agora a fase onde é descoberto o valor do *bit* seguinte, o décimo primeiro. A figura 2.8 seguinte mostra as ligações existentes nesta etapa.

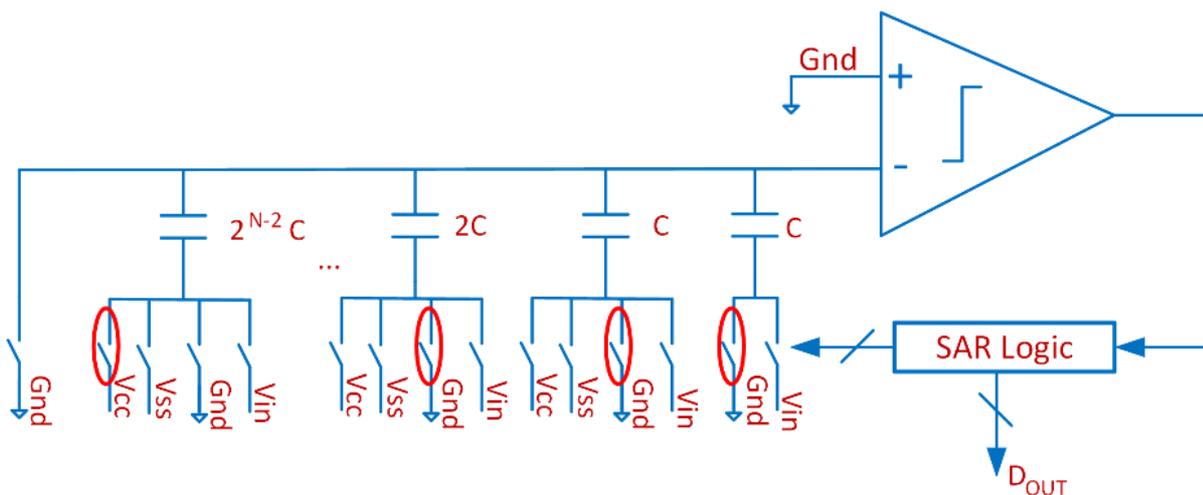


Figura 2.8 Comparação 11º bit

Comparativamente à fase anterior, a única diferença é a de que o condensador com maior capacidade está agora conectado a Vcc . De seguida, a figura 2.9 apresenta o circuito equivalente nesta altura do processo de conversão.

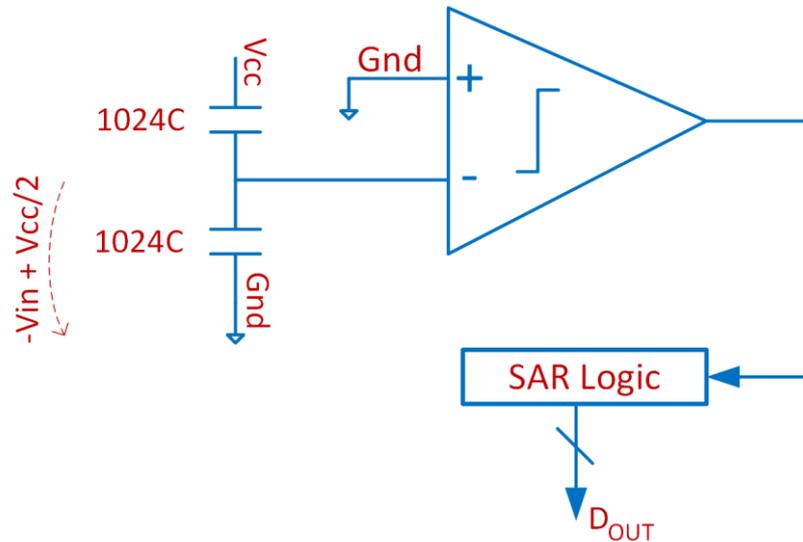


Figura 2.9 Comparação 11º bit (circuito equivalente)

Pela análise da figura 2.9, nota-se que ambos os condensadores perfazem um divisor de tensão de relação 1:1. Deste modo, à tensão que anteriormente estava presente na entrada negativa do comparador, será somada a tensão equivalente a $V_{cc}/2$.

No seguimento deste exemplo, a tensão presente na entrada negativa do comparador terá então o valor de -1.65 V ($-3.3 + 3.3/2 = -1.65$). Neste caso, como a tensão no terminal negativo do comparador é menor do que a tensão do terminal positivo, o 11º bit será também 1. É agora nesta fase que outra particularidade do algoritmo de conversão entra em vigor. Como o 11º bit tem o mesmo valor do que o 12º bit, o comutador que conectou o último condensador a V_{cc} permanece tal como está, sendo apenas mudado o estado dos comutadores que abordam o condensador com a seguinte maior capacidade, preparando assim a comparação para o próximo bit a descobrir. A figura 2.10 ilustra o estado dos comutadores nesta fase.

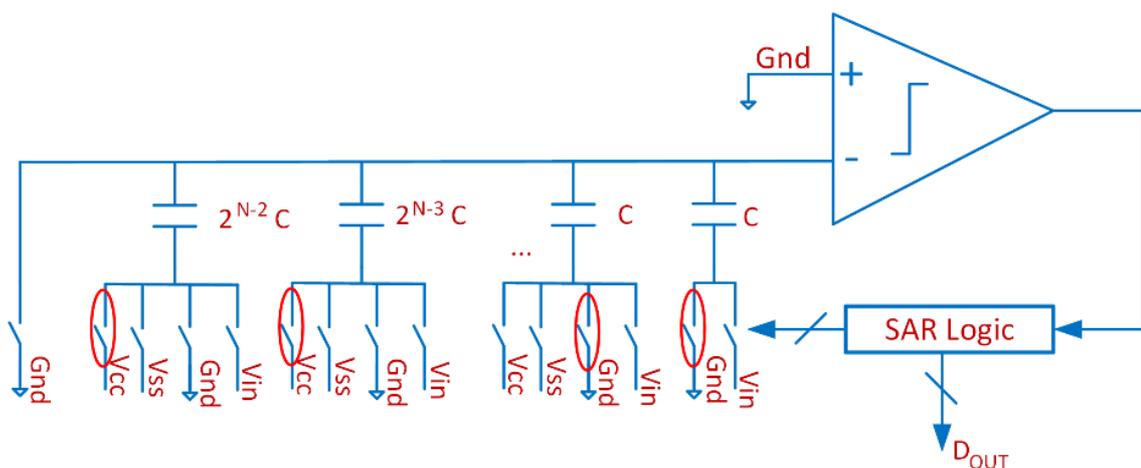


Figura 2.10 Comparação 10º bit

De acordo com o exemplo a ser estudado, o circuito equivalente nesta fase do processo apresenta-se na figura 2.11.

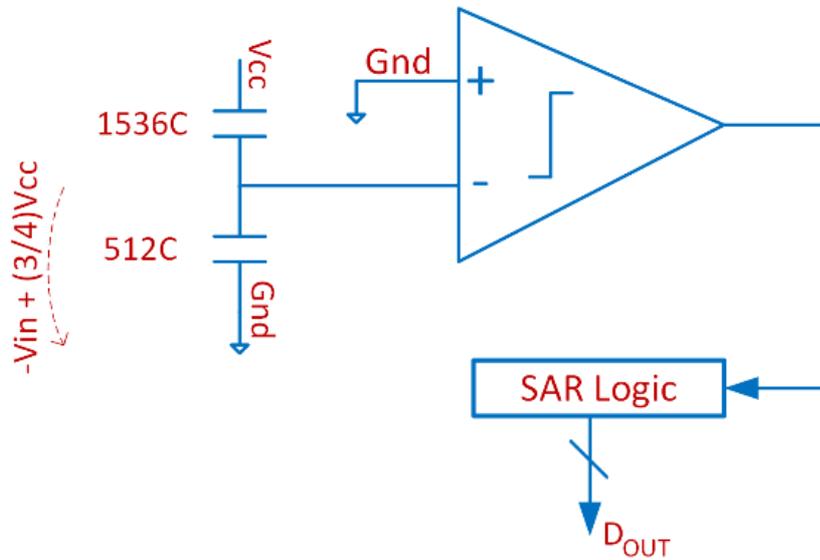


Figura 2.11 Comparação 10^o bit (circuito equivalente)

Nesta altura, a tensão presente no terminal negativo do comparador é -0.825 V ($-3.3 + (3/4)*3.3 = -0.825$). Como esta tensão ainda é menor que a tensão presente no terminal positivo do comparador, o resultado da comparação será novamente 1. De seguida, o algoritmo segue o mesmo raciocínio anteriormente descrito, ou seja, como o 10^o bit tem o mesmo valor que o 12^o bit, o último condensador que sofreu alterações permanece com a mesma configuração e o condensador que apresenta a seguinte maior capacidade é também ele conectado a V_{cc} . Este processo é repetido sucessivamente até que o último bit seja conseguido. Neste exemplo em particular, a palavra final seria composta por 12 bits todos com o valor 1 pois a tensão de entrada, V_{in} , tinha o valor mais alto passível de ser convertido ($3.3\text{ V} = V_{cc}$).

Recuando um pouco até ao circuito que permite descobrir o 10^o bit neste exemplo, assume-se desta vez um 11^o bit com o valor 0 em vez de 1. Nesta situação, visto que o 11^o bit tem um valor diferente do 12^o bit, o último condensador a ser conectado a V_{cc} vai passar ao estado correspondente ao início da conversão, ou seja, será novamente conectado a Gnd . Quanto ao condensador com a seguinte maior capacidade, ele é da mesma forma conectado a V_{cc} tal como no caso anteriormente descrito. A figura 2.12 demonstra o estado dos comutadores neste caso particular

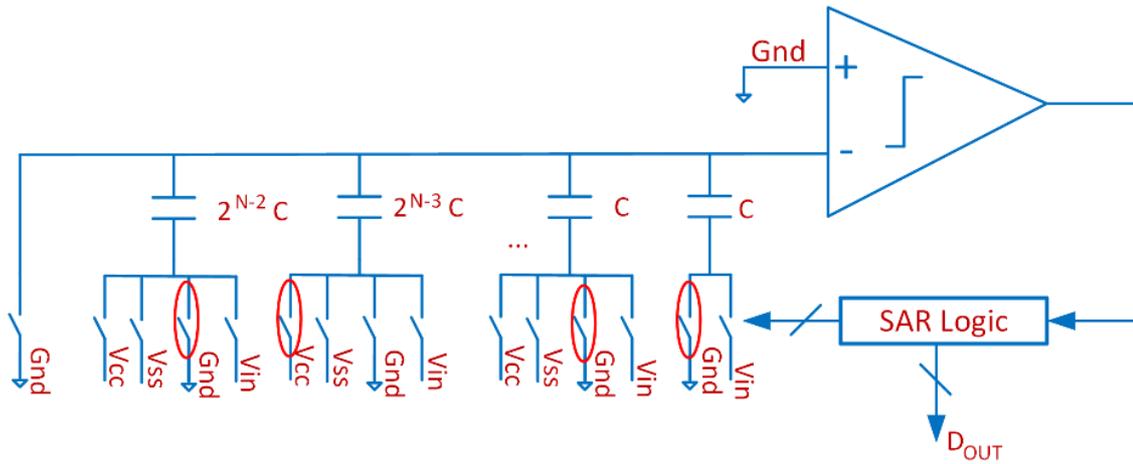


Figura 2.12 Comparação 10^o bit com 11^o bit com valor 0

O circuito equivalente que traduz a configuração que está representada na figura 2.12 é apresentado na figura 2.13.

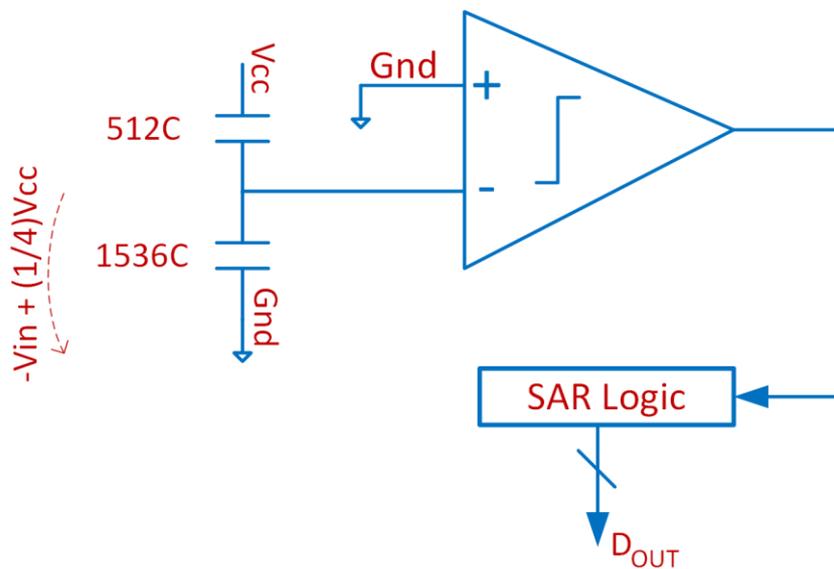


Figura 2.13 Comparação 10^o bit com 11^o bit com valor 0 (circuito equivalente)

Nesta fase, o processo é novamente repetido seguindo os mesmos princípios descritos até aqui até que o último *bit* seja descartado.

Apesar de todo este exemplo ser abordado com o 12º *bit* a 1, caso este tivesse o valor 0, o raciocínio a ser adotado seria exatamente o mesmo. A principal diferença é a de que os comutadores que permitem ligar os condensadores a V_{ss} seriam os manipulados em detrimento dos ligados a V_{cc} . No que diz respeito à manipulação dos comutadores, o algoritmo segue a mesma linhagem. Quando o *bit* que se descobre tem um valor diferente do 12º *bit*, neste caso 0, o último condensador a ser ligado a V_{ss} volta a ser conectado a Gnd e o condensador com a seguinte maior capacidade é ligado a V_{ss} e assim por diante até a conversão ficar concluída. Por outro lado, quando o *bit* descoberto tem o mesmo valor que o 12º *bit*, a única diferença para o caso anterior é a de que o último condensador a ser manipulado permanece da mesma forma conectado a V_{ss} .

3. SIMULAÇÃO DE ALTO NÍVEL

Segue-se agora a fase em que simulações de alto nível são realizadas e analisadas para posterior validação da arquitetura *SAR*. Esta trata-se de uma fase de extrema importância pois é nela onde pela primeira vez é testado e simulado todo o sistema que se pretende implementar. Como tal, devem ser feitos todos os testes necessários para que o processo de concretização do *ADC* possa prosseguir da forma mais clara e consistente possível.

Tendo em consideração as especificações impostas, este *ADC* terá 12 *bits* de resolução com uma alimentação de 3.3 V. Neste caso, o *bit* menos significativo representa uma variação de $\frac{3.3}{2^{12}} \approx 806 \mu V$ na tensão de entrada, ou seja, $V_{LSB} = 806 \mu V$. Este valor é uma referência bastante importante durante esta fase de simulação pois é com base nele que alguns dimensionamentos e análises serão realizadas.

3.1 Banco de condensadores ou *DAC*

A grande maioria dos conversores com arquitetura *SAR* utiliza um banco de condensares como parte fundamental do bloco anteriormente chamado de *DAC*, tendo este banco de condensadores um balanceamento binário. Devido a esta característica, a área ocupada pelo *DAC* varia exponencialmente com o número de *bits* do *ADC*, o que se pode revelar problemático.

Relembrando o exemplo de conversão apresentado em 2.4.5) que aborda um *ADC* com 12 *bits* como o que se pretende implementar, a razão entre o condensador de maior e menor capacidade era de 1024. Nesta fase de simulação, uma diferença de capacidades de tal ordem é de alguma forma admissível mas, tendo em conta uma fase mais avançada do processo, é totalmente inaceitável. Senão veja-se que, aquando da fase de *layout* mais concretamente, existiriam condensadores cuja área seria 1024 vezes maior do que a área ocupada pelo condensador com menor capacidade, acrescentado ainda o facto de que a esta situação ser-lhe-ia associada uma potência consumida também bastante elevada.

Devido aos fatores atrás mencionados, para que este obstáculo possa ser contornado, o *DAC* terá que sofrer algumas alterações de maneira a que o funcionamento do circuito não seja posto em causa. É sobretudo nesta reconfiguração do *DAC* que o próximo tópico se debruça.

3.1.1 *Split Capacitor arrays*

Uma forma astuta de reduzir a área ocupada que está diretamente relacionada a bancos de condensadores com balanceamento binário puro, é o uso dos denominados *split capacitor arrays*. Com o auxílio deste tipo de *arrays* de condensadores, não só a área ocupada diminui consideravelmente, como também a potência por estes utilizada.

Esta técnica consiste basicamente numa divisão dos bancos de condensadores com balanceamento binário puro em *arrays* mais pequenos, também balanceados, que se interligam entre si através de condensadores em série.

As figuras 3.1, 3.2 e 3.3 ilustram alguns arranjos possíveis, todos dimensionados para um *ADC* de 12 *bits*.

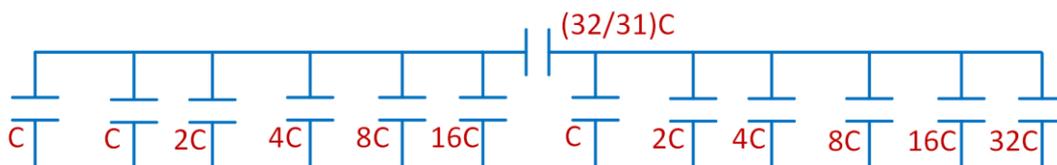


Figura 3.1 dois arrays balanceados com um condensador em série ($C_{total} \approx 96 * C$)

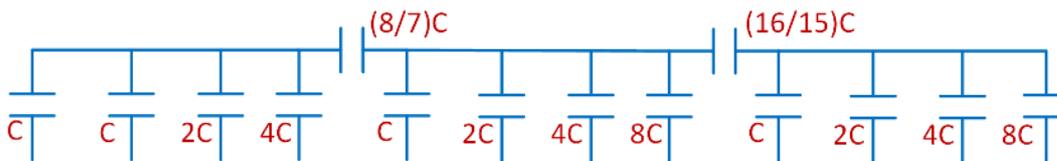


Figura 3.2 três arrays balanceados com dois condensadores em série ($C_{total} \approx 40 * C$)

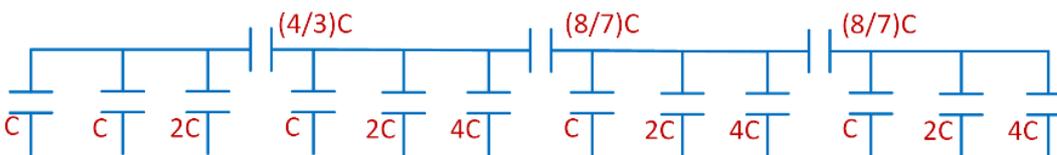


Figura 3.3 quatro arrays balanceados com três condensadores em série ($C_{total} \approx 28 * C$)

Tendo em conta as configurações anteriores, a abordagem onde existem apenas dois *arrays* balanceados apresenta uma capacidade total de aproximadamente $96 * C$. Quando comparado com as restantes configurações apresentadas, este valor é manifestamente elevado sendo por isso uma hipótese apenas considerada em caso de necessidade. Quanto às restantes, elas serão abordadas e analisadas mais à frente.

3.1.2 Dimensionamento dos condensadores

O dimensionamento dos condensadores que constituem o *DAC* é uma fase bastante importante pois existem alguns parâmetros que requerem atenção e, caso não sejam abordados da maneira correta, podem comprometer a performance de todo o sistema.

3.1.2.1 Condensador unitário

Segundo as possíveis configurações de *split capacitor arrays*, a maioria dos condensadores apresenta capacidades diferentes. Numa primeira análise, seria de esperar que o *DAC* seria constituído por condensadores com tamanhos diferenciados. No entanto, tal abordagem não é a mais indicada pois o uso exclusivo de condensadores unitários, ou seja, condensadores com capacidade C , revela uma melhoria na performance final [3]. Deste modo, e tomando como exemplo um condensador com capacidade $8 \cdot C$, este último seria equivalente a 8 condensadores unitários em paralelo.

3.1.2.2 Ruído kT/C

Outro aspeto importante é o ruído kT/C . Como o *DAC* terá na sua constituição condensadores unitários acompanhados dos respetivos comutadores, esta análise deve ser feita apenas a um condensador unitário bem como aos comutadores a ele conectados.

Apesar dos condensadores não geram qualquer tipo de ruído, estes acumulam ruído que é gerado por outras fontes. Neste caso em particular, a fonte responsável por gerar ruído será o *mosfet* ou *mosfets* que constituem o comutador, isto porque, quando ligados, o(s) *mosfet(s)* equivalem a resistências, elementos estes que são fontes de ruído.

Deste modo, o bloco a estudar pode representar-se como um simples circuito *RC*, sendo o seu ruído representado por $V_{noise(RMS)} = \sqrt{\frac{kT}{C}}$, $k = 1,3806488 \times 10^{-23} J/K$ [6].

Sendo este valor de tensão apenas o valor eficaz, a tensão de pico a pico terá um valor relativamente maior. Como tal, este valor deve representar apenas uma pequena parte do valor de V_{LSB} . O gráfico seguinte (figura 3.4) mostra a relação entre a tensão de ruído e o respetivo valor da capacidade, para uma temperatura de 403.15 K (130 °C).

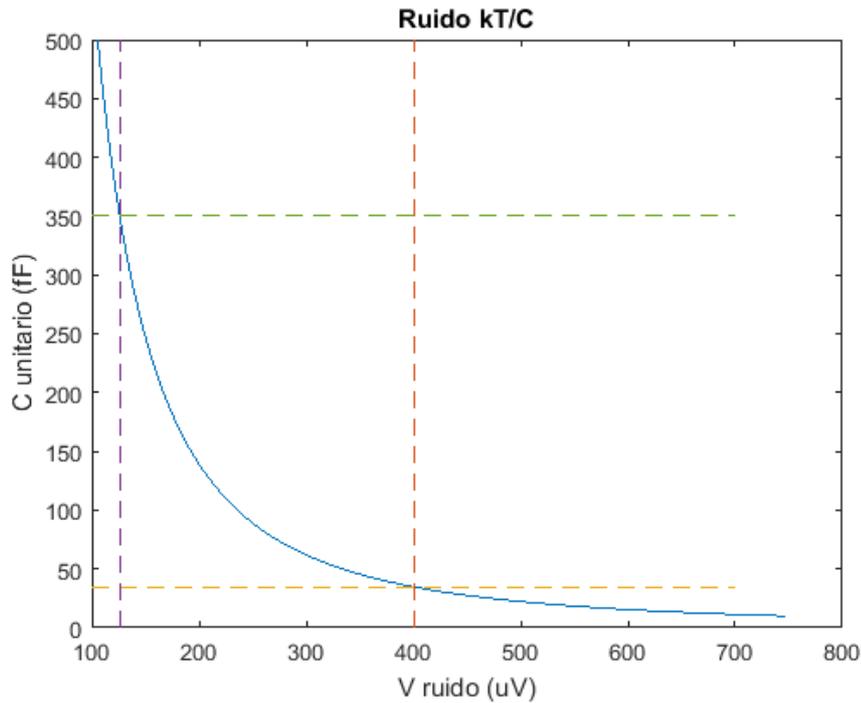


Figura 3.4 C unitário em função de V ruído

Para que o valor eficaz da tensão de ruído tivesse um valor de $400 \mu V$, aproximadamente $0.5V_{LSB}$, a capacidade do condensador unitário teria que ser sensivelmente $35 fF$. Este valor é manifestamente pequeno, especialmente se for tida em conta a tecnologia que se pretende utilizar.

Como tal, um condensador unitário de $350 fF$ de capacidade foi o escolhido, valor este que apresenta um valor eficaz da tensão de ruído de $126 \mu V$ ($\approx 0.15V_{LSB}$) a $130^\circ C$.

3.1.2.3 Estudo de tolerâncias

Nesta altura do processo, onde a capacidade do condensador unitário já se encontra definida, torna-se necessário estudar o comportamento do sistema nas mais variadas situações. Nesta perspetiva, o estudo das tolerâncias que o processo de fabrico apresenta é algo que não deve ser deixado ao acaso.

Posto isto, de entre todos os tipos de condensadores que o processo apresenta, escolheu-se o condensador do tipo *CPOLY* que, para uma capacidade de $350 fF$, necessita de aproximadamente $400 \mu m^2$. Em suma, o processo de fabrico apresenta uma tolerância de 0.03% dada a área ocupada pelo condensador.

Apesar do valor bastante baixo, é de extrema importância descobrir o seu efeito na precisão do *DAC*, especialmente quando existe mais do que uma possibilidade para a sua concretização.

Deste modo, é necessário analisar qual a situação que degradará mais a precisão deste bloco. Quase que de forma intuitiva, pode-se afirmar que a primeira comparação parece ser a mais problemática pois é nesta altura onde os condensadores com maior capacidade são manipulados, o que se poderá manifestar numa maior contribuição para a deterioração da precisão do DAC. Tal pensamento corrobora com a realidade pois o *bit* mais significativo é aquele que contribui em maior escala para a degradação do sinal de saída [7]. Tal como mencionado anteriormente, para que o DAC não ocupe uma área demasiadamente elevada, este bloco será constituído por sub *arrays* de condensadores balanceados, ligados entre si por resistências em série.

Nesse sentido, a primeira análise será feita com 3 *arrays* balanceados. A imagem 3.5 mostra as ligações a que cada um dos condensadores está sujeito nesta fase.

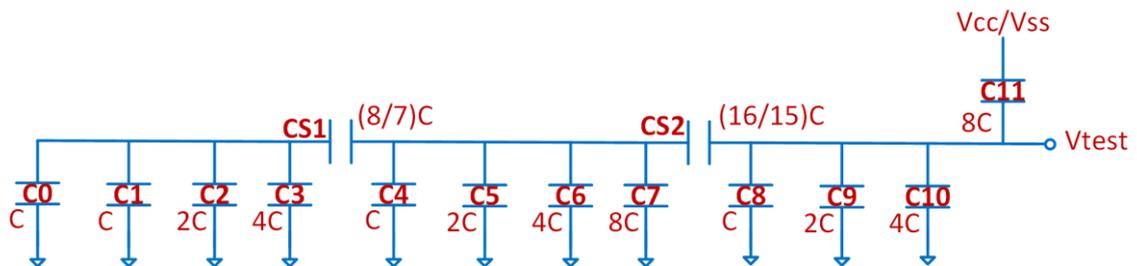


Figura 3.5 DAC com três arrays balanceados, na primeira fase de conversão

O raciocínio que deve ser adotado para conseguir chegar ao circuito equivalente do DAC aqui apresentado está explicado a seguir.

Em primeiro lugar, é necessário reduzir a um condensador equivalente o primeiro *array* de condensadores bem como o condensador em série a ele anexado. A expressão que o permite calcular apresenta-se a seguir.

$$C_{eq1} = \frac{[\sum_{k=0}^3 C_k] * C_{S1}}{[\sum_{k=0}^3 C_k] + C_{S1}} = \frac{8C * \frac{8}{7} * C}{8C + \frac{8}{7} * C} = C$$

Tendo em consideração o resultado da equação anterior, o circuito equivalente apresenta-se na figura 3.6.

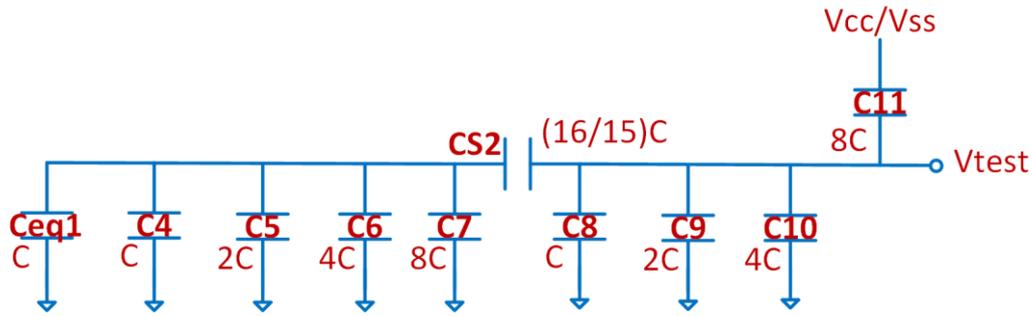


Figura 3.6 DAC com três arrays balanceados, na primeira fase de conversão (primeiro circuito equivalente)

Do mesmo modo, para que seja possível converter o segundo array em apenas um condensador equivalente, o raciocínio que se deve seguir segue exatamente o mesmo princípio.

$$C_{eq2} = \frac{([\sum_{k=4}^7 C_k] + C_{eq1}) * C_{S2}}{[\sum_{k=4}^7 C_k] + C_{eq1} + C_{S2}} = \frac{16C * \frac{16}{15} * C}{16C + \frac{16}{15} * C} = C$$

Nesta fase, o circuito equivalente resultante é apresentado na figura 3.7.

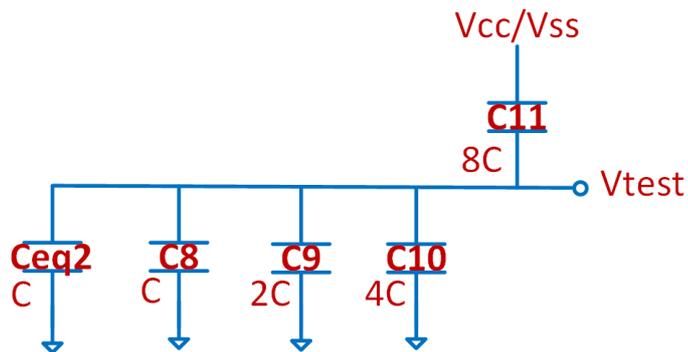


Figura 3.7 DAC com três arrays balanceados, na primeira fase de conversão (segundo circuito equivalente)

Pela configuração do circuito anterior, é notório que este não se encontra na sua forma mais simplificada. Somando as capacidades de todos os condensadores que se encontram em paralelo, é possível reduzir a constituição do circuito equivalente a apenas dois condensadores.

A figura 3.8 apresenta o circuito equivalente final.

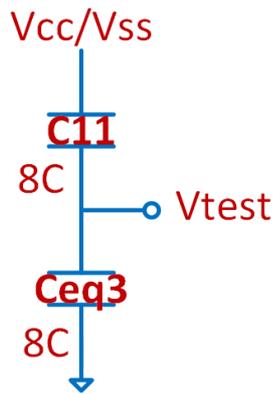


Figura 3.8 DAC com três arrays balanceados, na primeira fase de conversão (circuito equivalente final)

Segundo as especificações inicialmente impostas, apenas estará disponível uma tensão de 3.3 V bem como a respetiva massa, ou *gnd*. No que diz respeito ao funcionamento do DAC, são necessárias três tensões. Duas delas são exatamente o simétrico uma da outra, sendo a referência para estas duas últimas uma terceira tensão, *gnd*.

Nesta circunstância, para o bom funcionamento do DAC, será necessário gerar uma tensão que seja exatamente metade da tensão de alimentação, ou seja, 1.65 V. Neste caso, a tensão de 1.65 V funcionará como um *gnd* virtual do ponto de vista do DAC. Quanto às tensões de alimentação e *gnd*, elas representam para o DAC tensões de 1.65 V e -1.65 V, respetivamente.

Como tal, adotando esta nova terminologia de tensões, a tensão V_{test} assinalada no circuito equivalente final terá o valor de 0.825 V ou -0.825 V, dependendo da tensão a que o condensador superior está conectado. Nesta situação, onde todos os condensadores são perfeitos, a tensão presente em V_{test} é a ideal. Contudo o circuito final não apresenta apenas condensadores perfeitos.

Como referido no decorrer desta subsecção, a tolerância que o processo dita para condensadores com capacidade usada tem o valor de 0.03%, seguindo uma distribuição normal.

A figura 3.9 mostra a distribuição da diferença de tensão entre a tensão V_{test} ideal, 0.825 V, e a tensão V_{test} , proveniente de um DAC composto por condensadores não ideais. Esta análise foi realizada com um total de 5000 amostras.

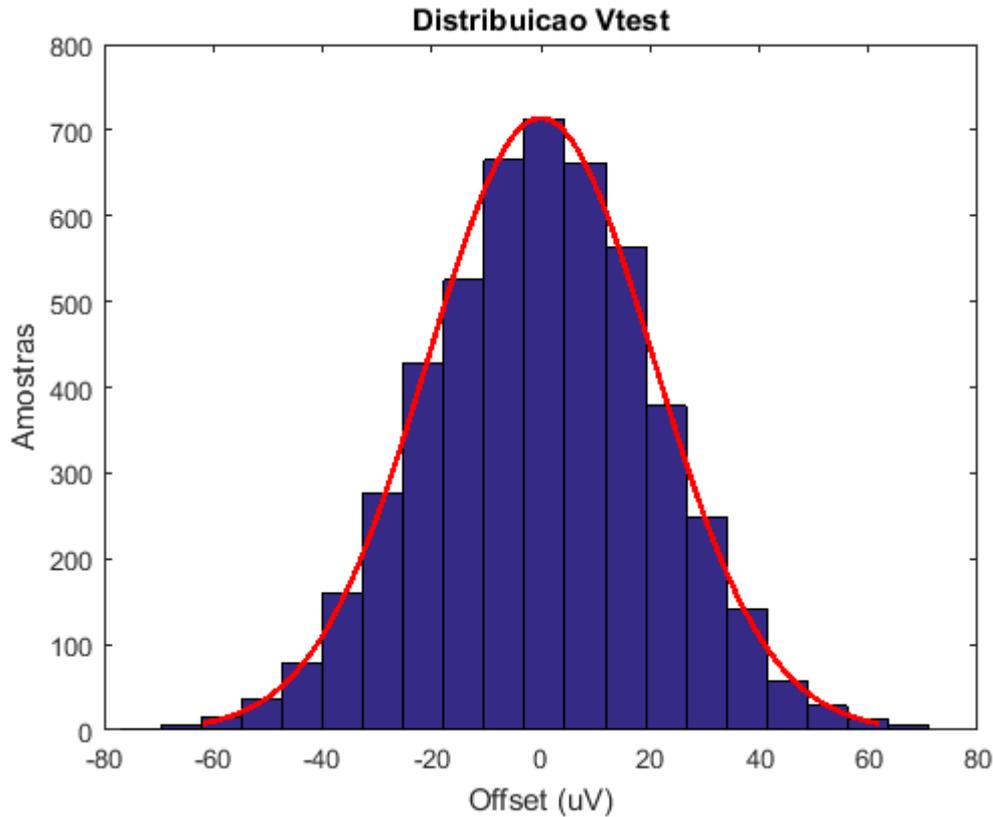


Figura 3.9 Distribuição de V_{test} em DAC com três arrays balanceados

Este estudo tem como resultados os apresentados a seguir:

Média: $-0.026579 \mu V$

Desvio padrão: $20.4193 \mu V$

3*Desvio padrão: $61.2579 \mu V$

Analisando mais profundamente os valores obtidos, e recorrendo para isso à análise estatística, a probabilidade de $V_{test} \in [Média - 3 * Desvio padrão, Média + 3 * Desvio padrão]$ é de 99.73%. Este resultado é bastante satisfatório, uma vez que 3*Desvio padrão representa apenas 8% de V_{LSB} . A conclusão final é a de que um DAC com três arrays balanceados constituído por condensadores unitários de $350 fF$ tem precisão suficientemente para um ADC de 12 bits.

Outra possibilidade que deve ser estudada é um DAC composto por quatro arrays balanceados. A figura 3.10 mostra a configuração do DAC em questão bem como as suas ligações.

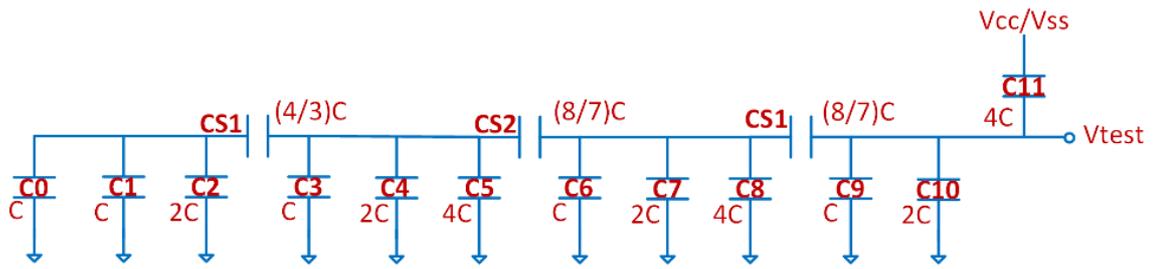


Figura 3.10 DAC com quatro arrays balanceados, na primeira fase de conversão

De maneira a conseguir um circuito equivalente mais simples, o mesmo raciocínio apresentado no DAC com três arrays de condensadores deve ser seguido. Como resultado, o circuito equivalente final é apresentado na figura 3.11.

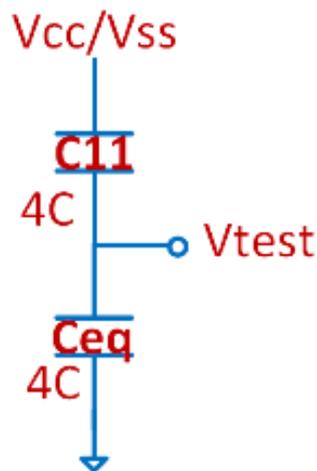


Figura 3.11 DAC com quatro arrays balanceados, na primeira fase de conversão (circuito equivalente final)

Do mesmo modo, a tensão V_{test} ideal tem também o valor de $0.825 V$, tal como no caso estudado anteriormente.

De seguida, na figura 3.12 é apresentada a mesma distribuição realizada no caso anterior, desta feita para um DAC com quatro arrays de condensadores balanceados.

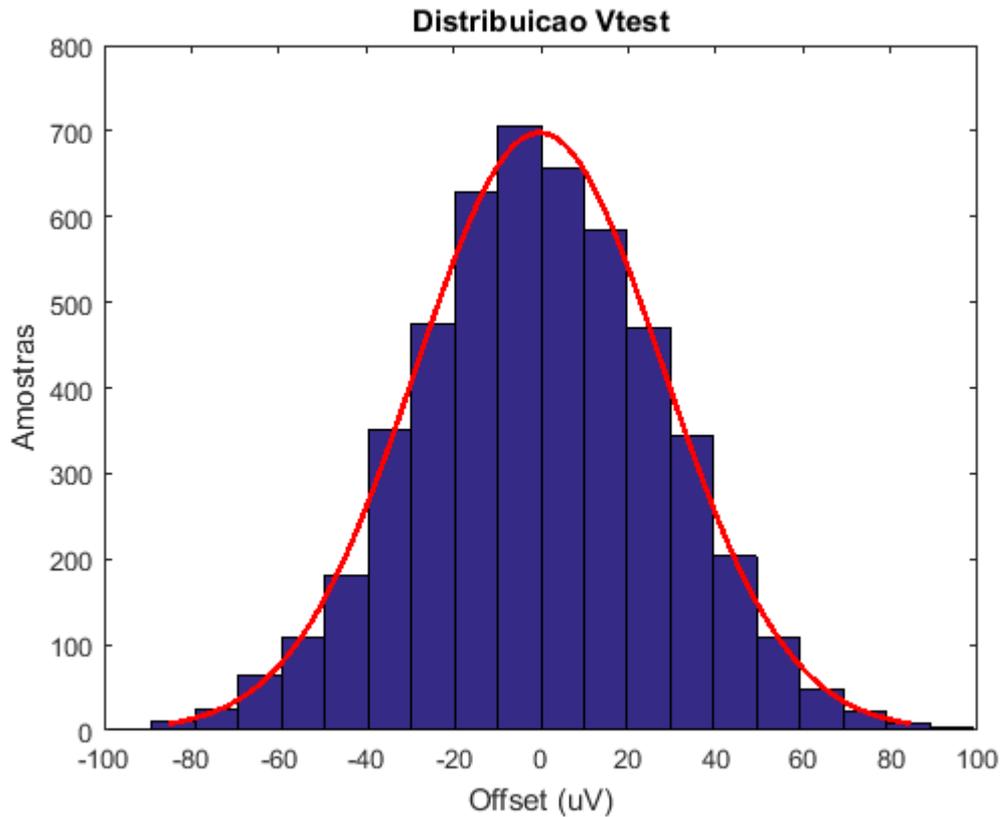


Figura 3.12 Distribuição de V_{test} em DAC com quatro arrays balanceados

Este estudo tem como resultados os apresentados a seguir:

Média: $-0.28728 \mu V$

Desvio padrão: $28.2862 \mu V$

3*Desvio padrão: $84.8586 \mu V$

Apesar do resultado proveniente deste teste ser ligeiramente pior do que o resultado anterior, é ainda assim um valor legítimo uma vez que 3*Desvio padrão representa apenas 10% de V_{LSB} .

Em jeito de conclusão, o DAC que apresenta maiores vantagens é o que é composto por quatro *arrays* de condensadores, sendo esta a utilizada no decorrer desta dissertação. Isto porque a vantagem de ter uma capacidade total mais baixa do que a outra possibilidade analisada suplanta a desvantagem de ter um desvio padrão ligeiramente superior.

É de notar que mais configurações para o DAC eram possíveis. No entanto, os ganhos não seriam muito vantajosos. Isto porque o desvio padrão iria aumentar, devido ao facto de se introduzir mais resistências em série, e os ganhos no que diz respeito à área ocupada não seriam significativos, uma vez que a redução das capacidades nos sub *arrays* de condensadores seria de certa forma suplantada pela necessidade de adicionar condensadores em série.

Este bloco apresenta no seu interface seis sinais de entrada, *Data1*, *Reset1*, *Vref*, *Vin*, *S/H*, e *Vm*, e uma de saída, *Out*.

Quanto ao primeiro sinal referido, *Data1* representa um barramento de 12 bits, sendo ele um dos responsáveis pela manipulação dos comutadores conectados aos condensadores. Os sinais *Reset1* e *S/H* são os sinais que, em conjunto com *Data1*, controlam toda a dinâmica envolvida nos comutadores. Por último, os sinais *Vref*, *Vm* e *Vin* dizem respeito às tensões de alimentação, tensão média (*gnd* virtual do *DAC*) e tensão de entrada, respetivamente.

3.2 Máquina de estados

Outro bloco que tem obrigatoriamente que ser parte integrante deste *ADC* é uma máquina de estados. Para que o *ADC* possa cumprir a sua função em tempo oportuno, é indispensável o auxílio de um bloco com as capacidades que uma máquina de estados apresenta.

A figura 3.14 ilustra a máquina de estados utilizada.

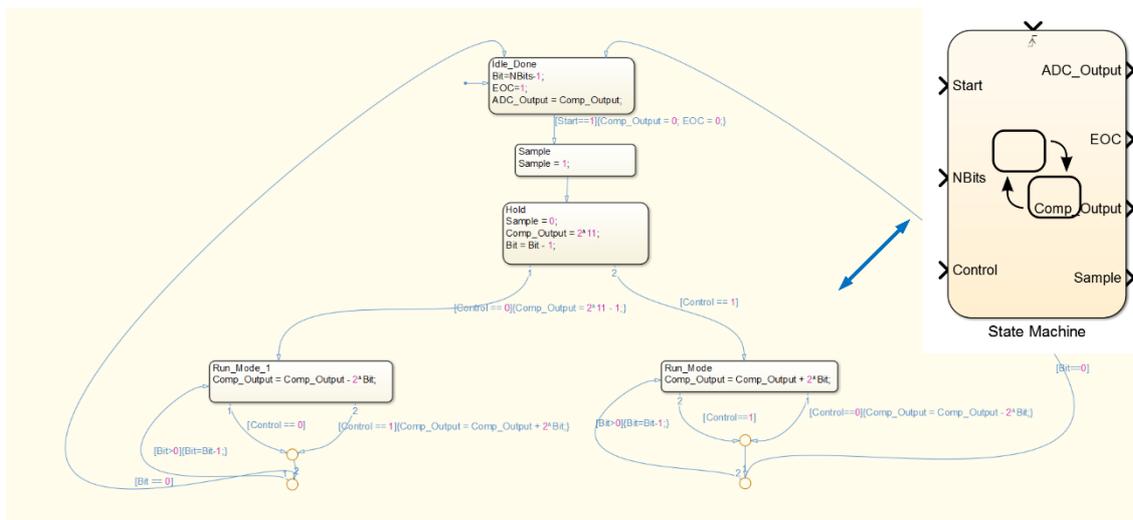


Figura 3.14 Representação da máquina de estados em Simulink

Tal como apresentado na figura anterior, cada uma das caixas negras representa um estado em que a máquina possa estar, sendo que em cada uma delas os sinais de saída apresentam valores referentes a esse mesmo estado. Quando uma transição de estado é requerida, as condições de teste são verificadas para que a máquina de estados possa transitar para o devido

estado de funcionamento. Estas transições estão representadas nas linhas azuis que interligam os vários estados.

Quanto ao seu interface, esta máquina de estados inclui quatro entradas, *Clk*, *Start*, *NBits* e *Control*, e também quatro saídas, *ADC_Output*, *EOC*, *Comp_Output* e *Sample*. O sinal de *Clk*, que se apresenta na parte superior do bloco, é o responsável por ditar à máquina de estados quando é que uma transição deve ser consumada, sendo o sinal de *Start* aquele que informa quando é que uma conversão deve ser iniciada. Quanto aos sinais *NBits* e *Control*, eles representam a resolução do *ADC* em *bits* e o resultado de uma comparação, respetivamente.

Passando agora para as saídas do bloco, o sinal *EOC* representa o final de uma conversão onde o barramento de saída *ADC_Output* é atualizado com o valor digital da respetiva conversão. No que diz respeito às duas restantes saídas, *Comp_Output* e *Sample* representam um barramento que controla os comutadores do *DAC* e um sinal que indica que a conversão está na fase de *Sample*, respetivamente.

3.3 Vista geral

Para que o *ADC* possa funcionar em total harmonia e sem qualquer tipo de problema, é necessário que todos os seus constituintes estejam em concordância e se possam interligar entre si sem comprometer a integridade do sistema. A figura 3.15 mostra a vista geral do sistema.

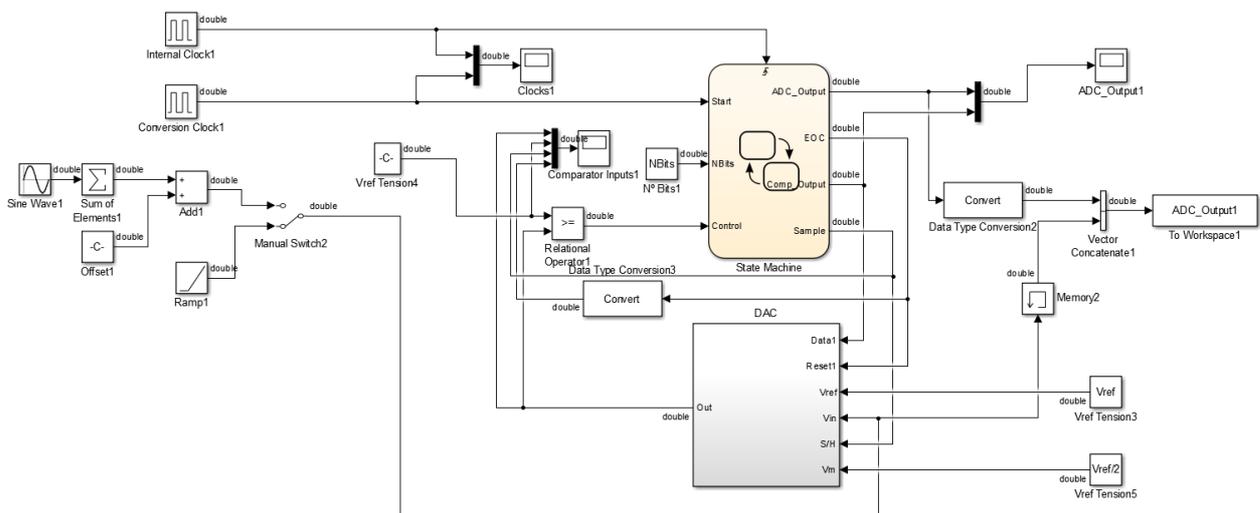


Figura 3.15 Representação do ADC em Simulink

Tal como mostra a figura anterior, tanto o *DAC* como a máquina de estados abordada anteriormente são parte integrante do sistema final. Outro bloco que também é muito importante

é o comparador que, visto apenas ser representado por um simples bloco de *Simulink*, não mereceu qualquer referência particular.

3.3.1 Análise de *INL* e *DNL*

Com a representação do *ADC* em *Simulink* totalmente operacional, nesta fase é possível recolher algumas características que o *ADC* pode adotar aquando do seu fabrico em silício. Duas dessas características são o *INL* e *DNL* [8].

Com o auxílio da geração aleatória de valores que o *Matlab* oferece, seguindo uma distribuição normal, tornou-se possível analisar qual a influência da variação das capacidades dos condensadores que resulta das tolerâncias que o processo de fabrico apresenta. Na figura 3.16 encontram-se ilustrados os valores de *INL* e *DNL* ao final de uma simulação.

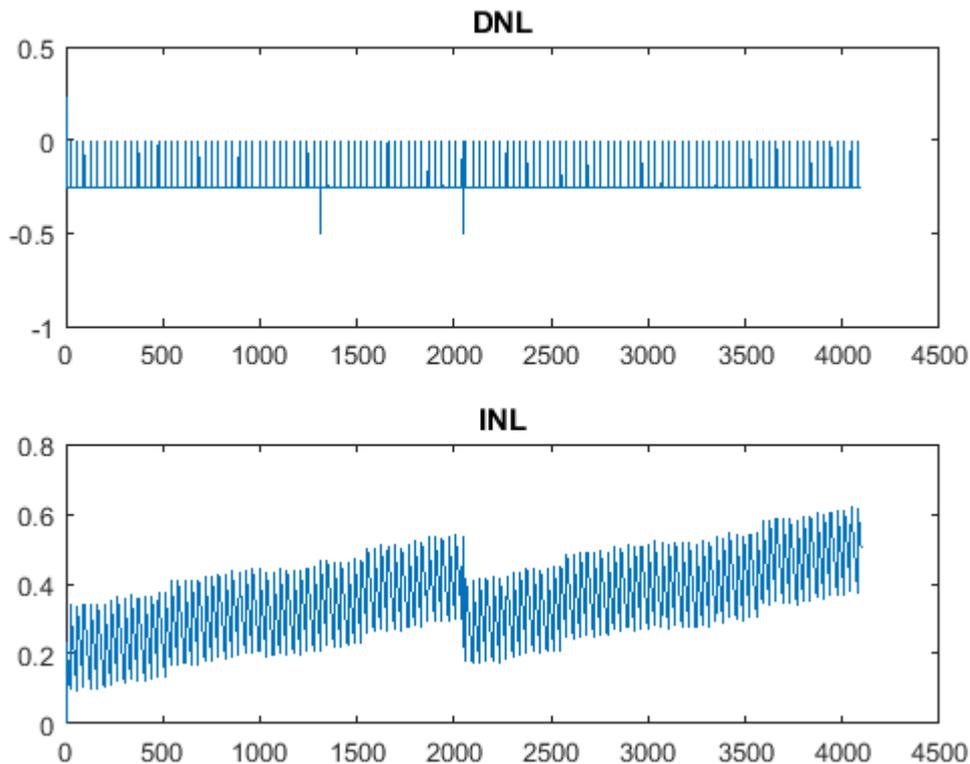


Figura 3.16 Valores de *DNL* e *INL* simulados

Os resultados apresentados na figura anterior são bastante satisfatórios uma vez que nenhum destes dois parâmetros ultrapassa, em módulo, o valor de V_{LSB} . Outro aspeto que revela alguma importância é o facto de não existirem *missing codes*. Por outras palavras, de todas as 2^N saídas que o *ADC* pode apresentar, também 2^N saídas diferentes foram conseguidas durante a simulação.

4. SIMULAÇÃO DE BAIXO NÍVEL

Depois de simulada, em alto nível, e aprovada a arquitetura do *ADC* que se pretende desenvolver, torna-se necessário baixar o nível de abstração, ou seja, dimensionar e simular todos os circuitos que constituem o *ADC* ao nível do *mosfet*.

Sendo assim, todos os blocos mencionados até agora têm que ter uma representação em baixo nível para que, numa fase final, todos possam ser convertidos em *layout* e posteriormente fabricados. Para tal, o uso das ferramentas do *Cadence* tornou-se vital uma vez que apresentam bastante versatilidade aliada a um grande poder de processamento de dados.

No decorrer deste capítulo, serão abordados todos os circuitos que constituem o *ADC* final, focando quais são as particularidades de cada um deles.

4.1 Comparador

O comparador é um dos elementos constituintes de um *ADC* deste tipo que revela uma maior importância. Isto deve-se ao facto dele ser o responsável por informar a máquina de estados quanto ao respetivo valor do *bit* da fase de conversão em questão. Outro aspeto que faz toda a diferença é a capacidade que o comparador deve ter em discriminar diferenças de tensão muito baixas, nunca ultrapassando o valor de V_{LSB} . No caso particular deste *ADC*, definiu-se que o comparador deve conseguir discriminar diferenças de tensão menores que $0.5V_{LSB}$, ou seja, cerca de $400 \mu V$.

No que toca a comparadores, a gama de escolhas é bastante vasta mas, visto que o mesmo circuito, quando sujeito a condições diferentes como a temperatura por exemplo, apresenta comportamentos distintos, o ideal passaria por poder controlar esse mesmo comportamento, independentemente das circunstâncias.

Como tal, um comparador apto a conseguir reduzir o seu próprio *offset* parece ser uma escolha bastante vantajosa dada a conjuntura atual. A figura 4.1 apresenta um comparador¹ dotado desta capacidade e descrito em [9].

¹ Informação adicional sobre o comparador em Anexo I - Comparador

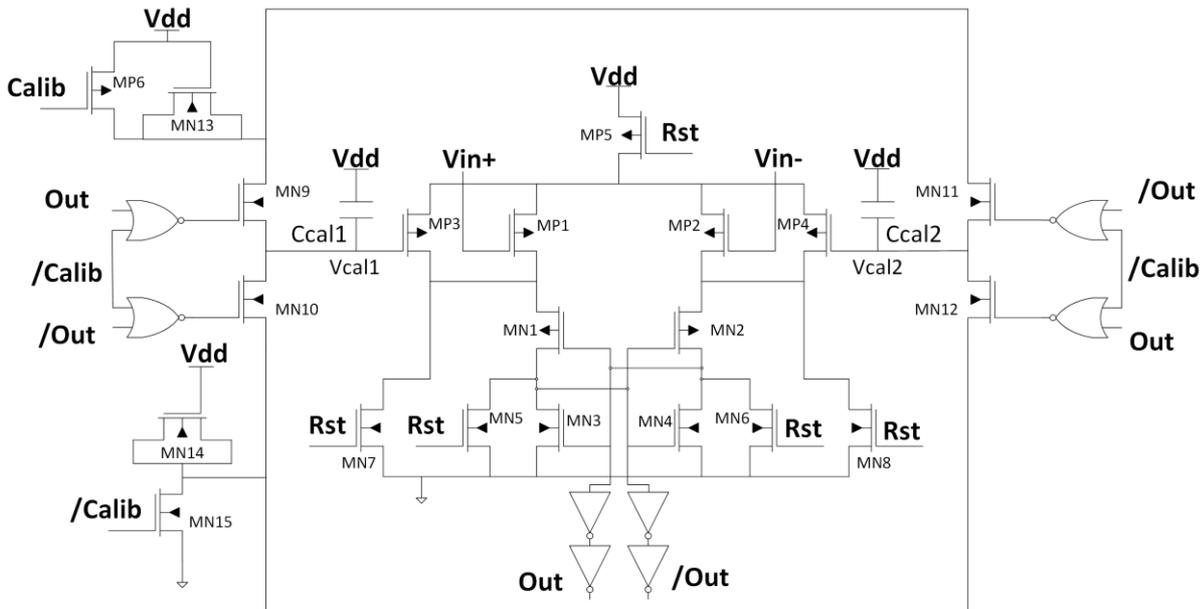


Figura 4.1 Comparador com circuito de auto-calibração (adaptado)

O circuito presente na figura anterior apresenta um modo de auto-calibração onde é possível reduzir o *offset* de entrada do comparador. Tal como descrito em [9], tal capacidade permite detetar variações de entrada na ordem dos $200 \mu V$. Este resultado é extremamente bom, uma vez que para este *ADC* está definido um *offset* de entrada máximo de $400 \mu V$.

4.1.1 Modo de auto-calibração

Este comparador em particular pode dividir-se em dois blocos fundamentais. O comparador (*track-and-latch*) em si, ou seja, o bloco que tem como única responsabilidade detetar a relação existente entre as duas entradas, e o bloco de auto-calibração, que tem como tarefa reduzir o *offset* de entrada do comparador.

Para que este último bloco consiga cumprir a sua função, o comparador tem que estar única e exclusivamente em modo de calibração. Portanto, nesta fase o *output* do comparador apenas tem significado para o bloco de auto-calibração a ele anexado, não tendo qualquer tipo de importância para qualquer outra identidade a ele conectado. Neste sentido, quando uma conversão está a decorrer, o comparador não pode entrar neste modo de funcionamento.

A solução passa então por, logo a seguir à comparação que dá a conhecer o último *bit* de uma conversão, reservar um espaço de tempo onde se possa proceder a uma calibração. De um modo geral, isto apenas se traduz na adição de mais um estado na dinâmica de conversão do *ADC*.

4.1.1.1 Princípio de funcionamento

O princípio de funcionamento inerente à auto-calibração deste comparador assenta fundamentalmente nas tensões V_{cal1} e V_{cal2} . Estas têm um papel preponderante, uma vez que têm influência na quantidade de corrente que atravessa ambos os ramos do comparador, o que influencia a sua decisão final reduzindo assim o *offset*.

Como anteriormente mencionado, a calibração só pode acontecer quando os 12 *bits* correspondentes à conversão em curso são conhecidos. No entanto, quando o *ADC* está a realizar conversões continuamente, não é recomendável a calibração no final de todas as conversões pois como estas têm um período definido, isso pode traduzir-se em harmónicos desagradáveis no sinal de saída do *ADC* [9]. A solução passa então por gerar um *bit* pseudo aleatório que dita quando é que uma calibração deve ou não ser realizada.

Apesar de ser necessário algum cuidado com o uso das calibrações, o seu funcionamento é sempre o mesmo. No início da mesma, é necessário que as tensões V_{in+} e V_{in-} tenham exatamente o mesmo valor. De seguida, procede-se a uma nova comparação que, consoante o seu resultado, irá ditar se o *offset* existente é positivo ou negativo uma vez que a mesma tensão está presente nas duas entradas do comparador. Para finalizar, o bloco de calibração utiliza o resultado da comparação feita imediatamente antes para que consiga decidir qual o procedimento que deve seguir. Nesta fase, existem apenas dois procedimentos possíveis, correspondendo cada um deles a um *offset* positivo ou negativo. Na primeira opção, a tensão V_{cal1} é aumentada ligeiramente sendo a tensão V_{cal2} diminuída no mesmo valor. Na restante alternativa, acontece exatamente o oposto.

Quando uma calibração é terminada, o *offset* sofre apenas uma ligeira redução visto que as mudanças tanto em V_{cal1} como em V_{cal2} são também elas diminutas. No caso de existir um *offset* relativamente elevado, é necessário proceder a várias calibrações até que o *output* do comparador comece a alternar o seu valor entre 0 e 1. Quando isto acontece, significa que o bloco de calibração já reduziu o *offset* do comparador até ao mínimo possível.

4.1.2 Tempo de estabilização do *output*

Outro aspeto que deve ser analisado neste comparador é o tempo que ele necessita para que o seu *output* estabilize. Visto que as especificações iniciais para este *ADC* são 12 *bits* de resolução e uma taxa de amostragem de 3 *MSPS*, então a máquina de estados, para que uma conversão seja possível, terá que contemplar 15 estados. Os dois estados iniciais dizem respeito

à fase de *Sample* e de *Hold*, os doze seguintes representam cada um dos *bits* de resolução do *ADC* e, por último, um estado reservado para a calibração do comparador.

No que à frequência diz respeito, para que seja possível uma taxa de amostragem de 3 *MSPS*, a máquina de estados precisa de um *clock* com uma frequência tantas vezes superior à taxa de amostragem quantos os estados que a máquina de estados necessita para realizar uma conversão, ou seja, 15. Logo, a máquina de estados necessita de um *clock* de 45 *MHz*.

Dado que a frequência de relógio já é conhecida, e visto que o resultado da comparação tem que estar disponível em apenas meio ciclo de relógio (consultar tópico 4.4.2), então o comparador só dispõe de $\frac{1}{2} * \frac{1}{45 \text{ MHz}}$, ou seja, cerca de 11.1 *ns*. Neste sentido, realizaram-se simulações para cada um dos denominados *process corners*² às temperaturas de -30°C, 30°C e 130°C.

A figura 4.2 mostra os resultados obtidos.

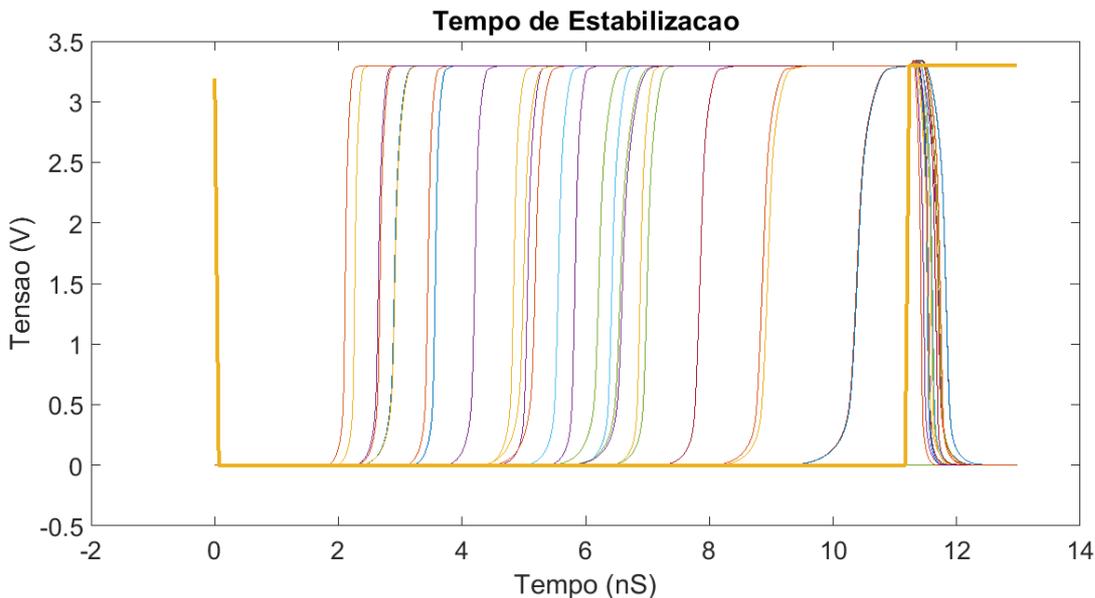


Figura 4.2 Tempo de estabilização da saída do comparador

Na imagem anterior, o traço amarelado que se encontra destacado representa o sinal de *clock*, estando apenas mostrado meio do seu ciclo a uma frequência de 45 *MHz*. Quanto aos restantes sinais, eles mostram qual a evolução da saída do comparador para cada uma das simulações realizadas.

Quando o sinal de *clock* está na sua transição descendente, o processo de comparação por parte do comparador é iniciado e, devido à dinâmica imposta pela máquina de estados, o

² Informação adicional sobre *process corners* em Anexo II – *Process corners*

resultado da comparação terá que estar disponível antes da próxima transição ascendente do sinal de relógio. Pela análise dos resultados, esse requisito foi cumprido em todos os casos estudados.

É também importante referir que todas as simulações foram realizadas com valores de V_{in+} e V_{in-} idênticos, isto porque durante as simulações verificou-se que quando as entradas apresentam valores muito semelhantes o tempo de estabilização do resultado da comparação aumenta ligeiramente.

4.1.3 Análise de *offset*

Visto que a tecnologia de fabrico utilizada em [9] é diferente da tecnologia especificada no início desta dissertação, torna-se necessário analisar qual a performance que este comparador apresenta em circunstâncias diferentes. Como tal, para estudar o seu comportamento foi utilizada a configuração apresentada na figura 4.3.

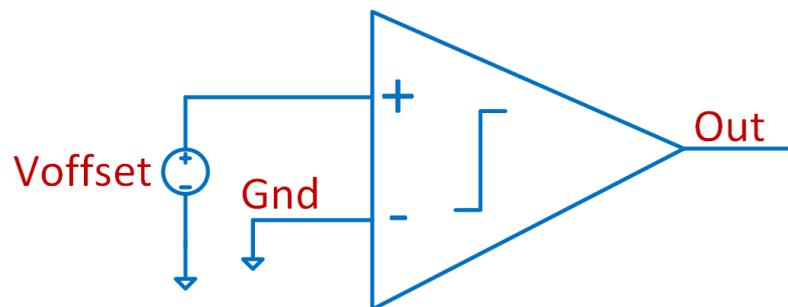


Figura 4.3 Configuração do comparador para estudo do *offset* de entrada

Para esta análise, foram feitos testes com tensões de *offset* de 5 mV , 10 mV , -5 mV e -10 mV . Tal como no estudo do tempo de estabilização da saída do comparador, também aqui foram contemplados os *process corners*, cada um deles simulado para as temperaturas de -30°C , 30°C e 130°C .

A figura 4.4 revela os resultados obtidos.

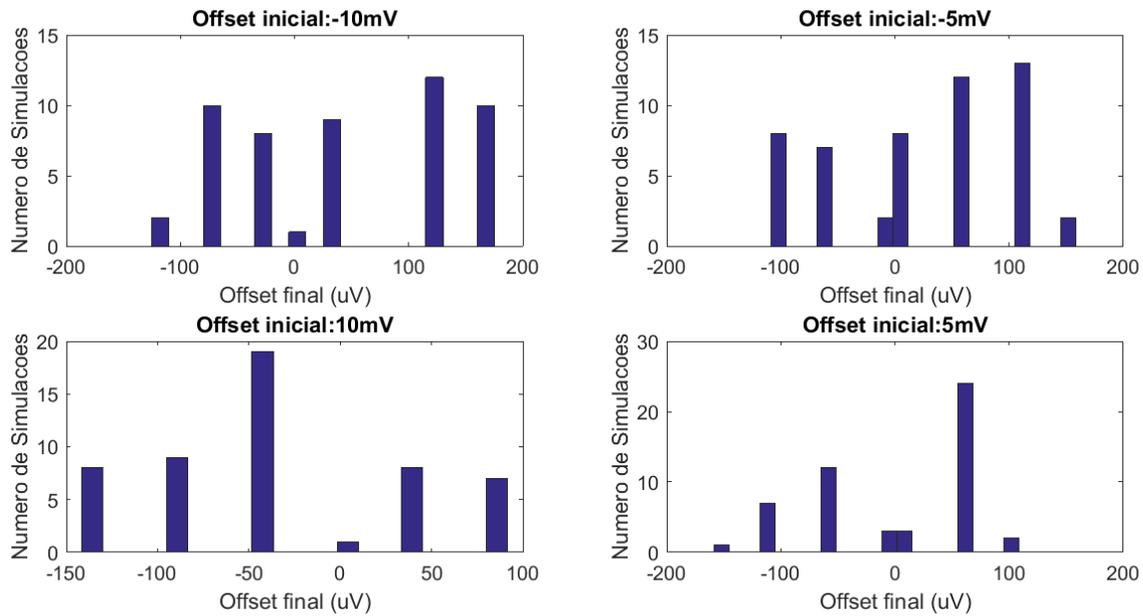


Figura 4.4 Offsets obtidos depois da calibração

Segundo os resultados conseguidos (figura 4.4), tal como no tempo de estabilização, também o *offset* de entrada cumpriu os requisitos impostos, não ultrapassando os estipulados $400 \mu V$.

4.2 Amplificador operacional

Outro elemento que deve fazer parte deste *ADC* em específico é um amplificador operacional. Apesar de este não ser um bloco fundamental enumerado anteriormente, ele é necessário visto que o *DAC* necessita de uma tensão intermédia para o seu bom funcionamento. Como tal, o dimensionamento de um amplificador operacional revelou-se algo essencial para a integridade de todo o sistema.

Posto isto, a configuração do amplificador operacional³ utilizado encontra-se na figura 4.5 [6], acrescido de *mosfets* de *power-down*.

³ Informação adicional sobre amplificador em Anexo III – Amplificador operacional

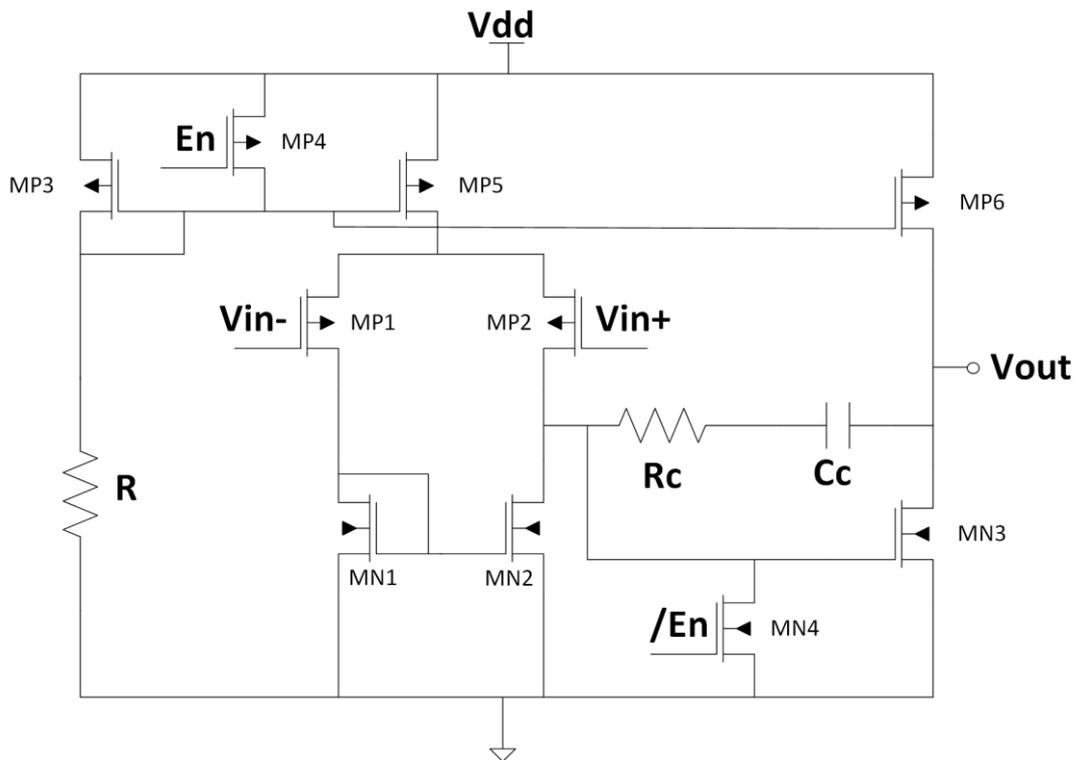


Figura 4.5 Amplificador operacional (adaptado)

Este amplificador será utilizado numa topologia de seguidor de tensão⁴, tensão essa com proveniência de um divisor de tensão de relação 1:1 (figura 4.6), alimentado pela tensão de alimentação, ou seja, 3.3 V. Deste modo, o amplificador terá à sua saída a tensão de 1.65 V que o DAC necessita para funcionar.

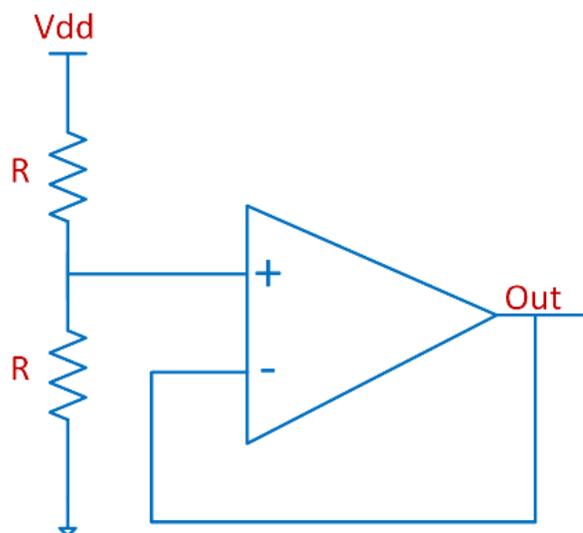


Figura 4.6 Amplificador configurado como seguidor de tensão

⁴ Informação adicional sobre a ligação em seguidor de tensão em Anexo III – Amplificador operacional

4.2.1 Variação da tensão de entrada

Nas condições ideais de funcionamento, a tensão intermédia não apresenta qualquer tipo de desvio mas, devido às limitações do amplificador, na realidade a tensão apresenta algumas diferenças em relação ao valor ideal. Isto deve-se sobretudo ao facto do amplificador ter um ganho limitado. Aliado ainda ao ganho limitado, tal como anteriormente mencionado, os *process corners* influenciam o funcionamento de qualquer circuito e, conseqüentemente, também o ganho sofre alterações dependendo do *process corner* em questão.

Depois de dimensionado o amplificador e de protagonizados alguns ajustes no decorrer das simulações, foi conseguido, no caso típico de funcionamento, um ganho de aproximadamente 64 dB. A figura 4.7 mostra as diferenças que a tensão de saída do amplificador apresenta em relação à tensão ideal (1.65 V) nos diferentes *process corners*.

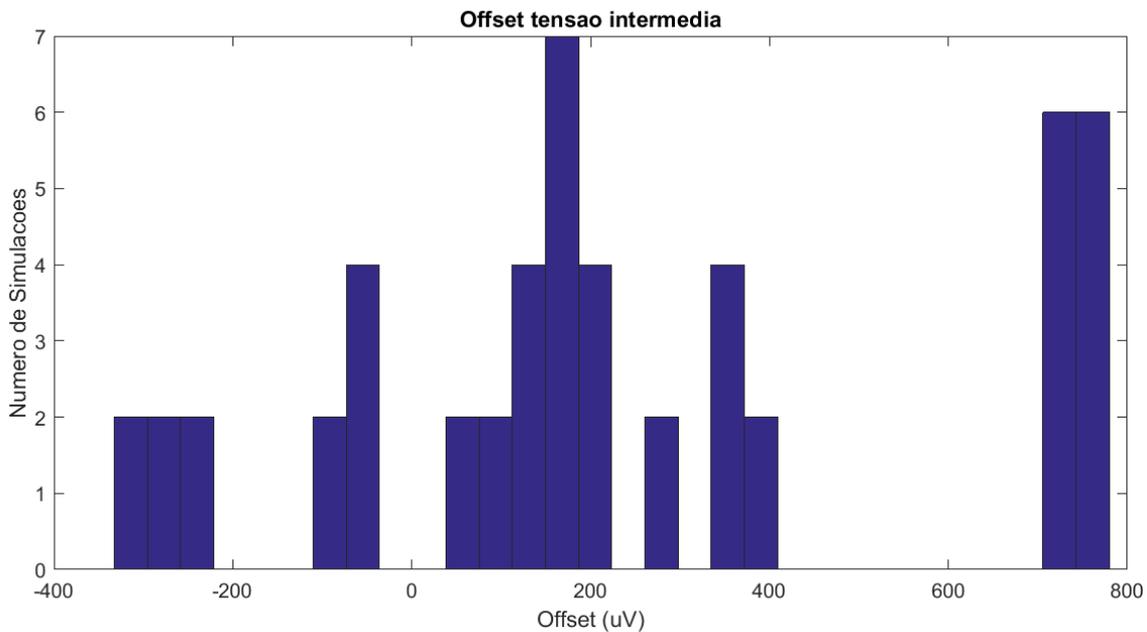


Figura 4.7 Diferença entre tensão ideal e saída do amplificador

Tal como se pode verificar pela figura, em grande parte das simulações a diferença não excedeu 50% do valor de V_{LSB} , apesar de em alguns casos esta diferença atingir um valor muito próximo de V_{LSB} . No cômputo geral, os resultados são satisfatórios uma vez que o desvio máximo não ultrapassa, em módulo, a tensão V_{LSB} .

4.2.2 Análise em frequência

Outro aspeto que deve ser abordado é o comportamento do amplificador no domínio da frequência. Apesar de este amplificador ser única e exclusivamente utilizado como seguidor de tensão, esta análise revela bastante importância uma vez que, quando é solicitada corrente ao amplificador, a sua saída tem um tempo limitado para estabilizar, como explicado mais adiante.

4.2.2.1 Análise em malha aberta

Um dos primeiros aspetos a serem analisados em qualquer amplificador é verificar a sua estabilidade e respetiva margem de fase. Para tal, a forma mais indicada para o realizar é proceder à análise em frequência, utilizando o diagrama de Bode que o amplificador produz. A figura 4.8 mostra os resultados obtidos neste amplificador.

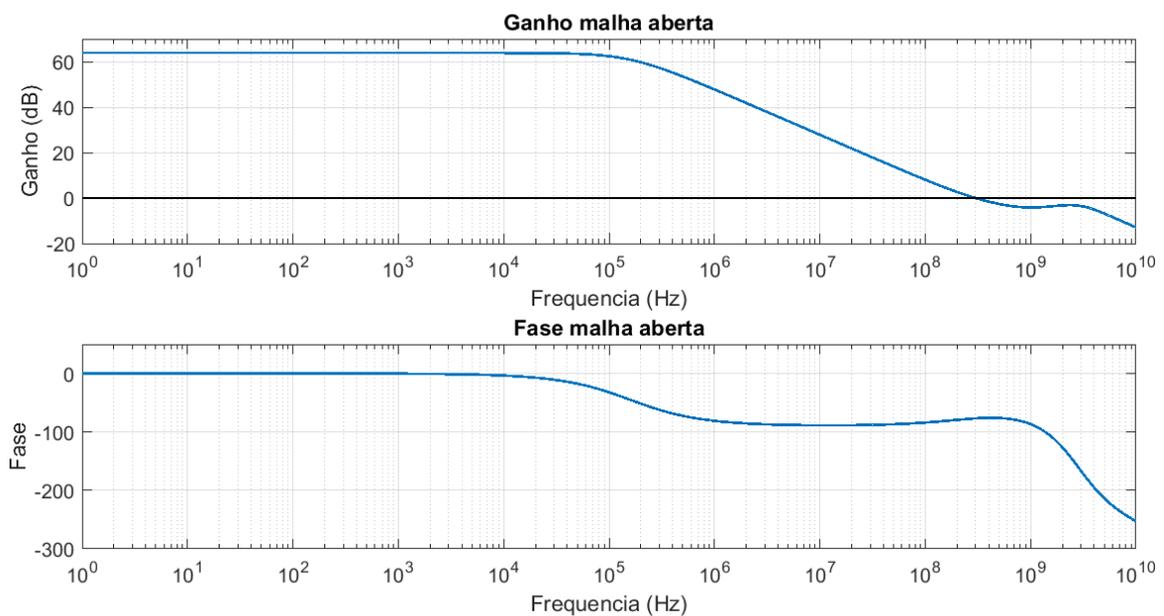


Figura 4.8 Diagrama de Bode do amplificador em malha aberta

Pela análise do diagrama de Bode, conclui-se que este amplificador é estável pois, quando o seu ganho é unitário, a sua fase não ultrapassa, em módulo, os 180° [6]. Quanto à margem de fase, ela é aproximadamente 90° .

4.2.2.2 Análise em malha fechada

Visto que o amplificador apresentado é estável, segue-se a análise em malha fechada. Neste caso particular, analisou-se o comportamento do amplificador em frequência quando configurado como seguidor de tensão. A figura 4.9 ilustra o diagrama de Bode para esta configuração.

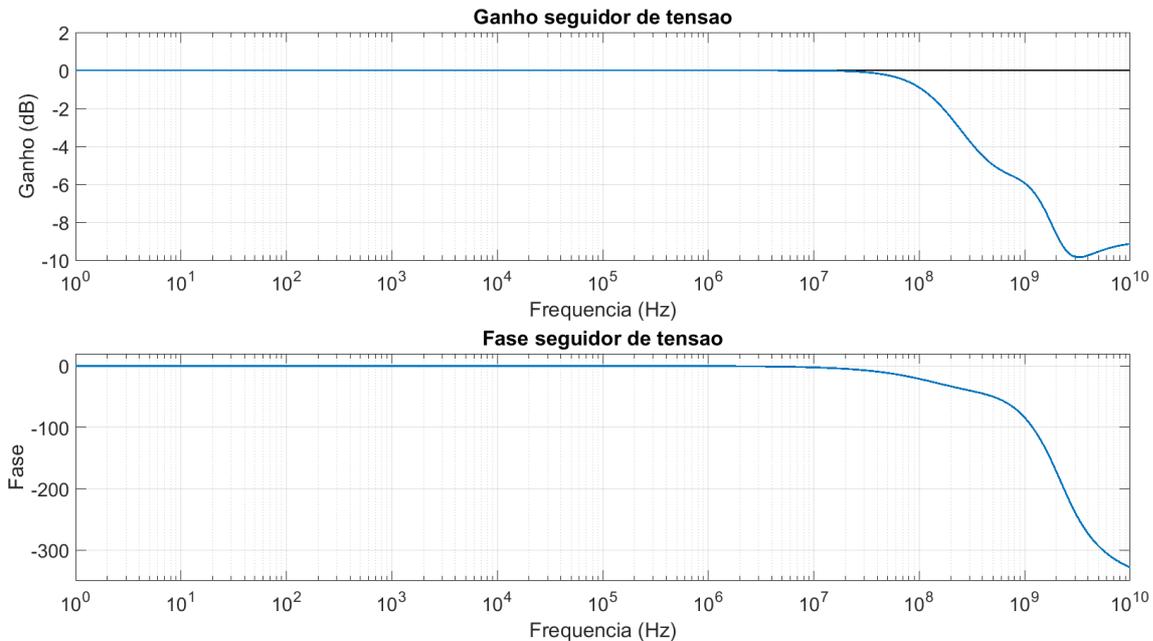


Figura 4.9 Diagrama de Bode para amplificador configurado como seguidor de tensão

Tal como explicado na secção referente ao comparador, para que a taxa de amostragem imposta inicialmente não seja comprometida, a máquina de estados necessita de um sinal de *clock* de 45 MHz. Devido a este valor conjuntamente com a dinâmica imposta pela máquina de estados (ver secção 4.4 - máquina de estados), a saída do amplificador deve estabilizar em meio ciclo de relógio, ou seja, aproximadamente $11 \text{ ns} \left(\frac{1}{2} * \frac{1}{45 \text{ MHz}} \right)$.

Pelo estudo do diagrama de Bode, pode-se afirmar que este amplificador, quando configurado como seguidor de tensão, consegue colocar na sua saída sinais até sensivelmente 100 MHz sem qualquer tipo de atenuação em relação ao sinal de entrada. Apesar de este amplificador estar apenas sujeito a uma entrada com uma tensão contínua (1.65 V, 0 Hz), a sua saída vai oscilar quando lhe é pedido corrente elétrica. Essas situações ocorrem aquando da reorganização dos comutadores que constituem o DAC, procedimento este que obriga a transferências de carga entre os condensadores e consequentes picos de corrente.

Com toda a informação recolhida até este ponto, pode proceder-se a uma aproximação que possa de certo modo prever qual o tempo que o amplificador demora a estabilizar a sua saída. Tendo em conta os 100 MHz que o amplificador consegue colocar à saída sem atenuação, pode-se de certo modo dizer que o amplificador vai demorar tanto tempo a estabilizar quanto o período correspondente aos 100 MHz , ou seja, 10 ns . Este valor cumpre com os requisitos visto que a sua saída deve estabilizar em menos de 11 ns . É importante referir que, recorrendo às análises das simulações, as constantes de tempo protagonizadas pelas capacidades dos condensadores e resistências dos comutadores que constituem o *DAC* eram pequenas o suficiente para que tal aproximação fosse considerada.

4.3 Banco de condensadores ou *DAC*

O banco de condensadores é provavelmente o elemento integrante de um *SAR ADC* que merece um estudo mais aprofundado e maior atenção uma vez que é este bloco o responsável por gerar tantas tensões de comparação quantos os *bits* de resolução do *ADC*. Tal como abordado no capítulo da simulação de alto nível, existem várias configurações que podem ser usadas, cada uma delas com os seus pontos mais ou menos fortes, e que merecem ser analisadas tendo sempre em conta as especificações impostas pelo sistema a desenvolver.

Neste caso particular, optou-se por um *DAC* constituído por quatro *arrays* de condensadores balanceados com três condensadores em série. A figura 4.10 mostra de forma mais detalhada qual a constituição deste bloco.

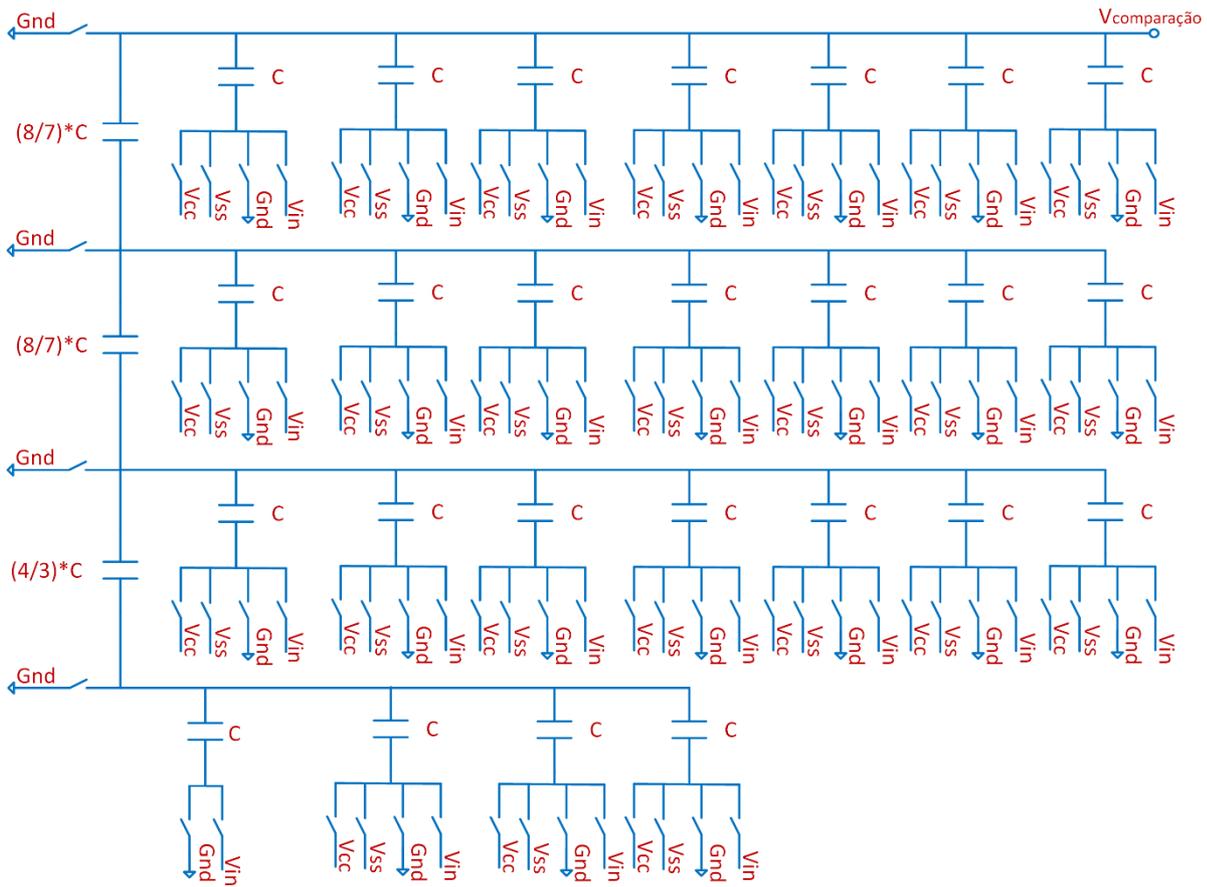


Figura 4.10 Configuração do DAC

Tal como ilustrado na figura 4.10, a grande maioria dos condensadores que constituem o *DAC* têm, no seu terminal inferior, a eles conectados quatro comutadores que permitem escolher entre quatro tensões diferentes. No que diz respeito à tensão *Gnd* apresentada na figura, ela representa a massa virtual do *DAC*, ou seja, a tensão que o amplificador mencionado no tópico anterior providencia. Quanto às três restantes tensões, *Vin* representa a tensão que se pretende converter e *Vcc* e *Vss* representam as tensões de 3.3 V e 0 V, respetivamente. Quanto aos comutadores que permitem ligar ambas as extremidades dos condensadores em série à massa virtual, o seu papel revela-se essencial pois não só permitem descarregar os condensadores em série como permitem carregar todos os condensadores constituintes dos quatro *arrays* de condensadores com a tensão que se pretende converter, aquando da fase de *Sample*.

4.3.1 Dimensionamento dos condensadores

Visto que a capacidade escolhida para os condensadores unitários tem o valor de 350 fF , traduzidos numa área de $400\text{ }\mu\text{m}^2$, então o condensador pode adotar uma forma quadrada com um lado de $20\text{ }\mu\text{m}$. Quando introduzidas estas medidas no *software* de simulação, a capacidade efetiva do condensador tem o valor de 350.88 fF . Tal como os condensadores em paralelo, também o dimensionamento dos condensadores em série seguiu o mesmo procedimento, culminando em condensadores de $21\text{ }\mu\text{m}$ por $21.8\text{ }\mu\text{m}$ ($\frac{8}{7} * C_{\text{unitário}}$) e $22.75\text{ }\mu\text{m}$ por $23.5\text{ }\mu\text{m}$ ($\frac{4}{3} * C_{\text{unitário}}$).

4.3.2 Dimensionamento dos comutadores

Apesar dos condensadores serem os elementos com maior destaque neste bloco, os comutadores que a eles estão ligados, caso mal dimensionados, podem prejudicar em larga escala o desempenho do *DAC*. Como tal, os *mosfets* a serem utilizados como comutadores não só necessitam de um tamanho físico como também precisam que o seu tipo, n ou p, seja selecionado da forma mais adequada.

Começando pelos comutadores que permitem conectar os condensadores a V_{cc} e V_{ss} , eles são, respetivamente, do tipo p e n. O comutador que liga a V_{cc} é representado por um *mosfet* do tipo p porque, para este entrar em condução, a tensão V_{GS} tem que ser menor do que V_{th-p} do *mosfet* em questão. Visto que V_{cc} tem um valor de 3.3 V e está ligado à *source* do *mosfet*, este entra em condução quando ao terminal *gate* são aplicados 0 V , fazendo assim com que $V_{GS} = 0 - 3.3 = -3.3\text{ V} < V_{th-p}$. No caso de à *gate* serem aplicados 3.3 V , o *mosfet* não conduz pois $V_{GS} = 3.3 - 3.3 = 0\text{ V}$.

Pelo contrário, o comutador que liga a V_{ss} (0 V) é representado por um *mosfet* do tipo n pois a tensão V_{GS} tem que ser maior do que V_{th-n} para que o *mosfet* entre em condução. Neste caso, quando ao terminal *gate* são aplicados 3.3 V , e relembrando os 0 V do terminal *source*, faz com que $V_{GS} = 3.3 - 0 = 3.3\text{ V} > V_{th-n}$. Neste *mosfet* em particular, quando aplicada à *gate* uma tensão de 0 V , não existe condução uma vez que $V_{GS} = 0 - 0 = 0\text{ V}$.

Quanto ao comutador que permite ligar o terminal inferior do condensador a Gnd (massa virtual do *DAC*), ou seja, 1.65 V , ele traduz-se num *mosfet* do tipo n. Apesar de um do tipo p também surtir efeito neste caso, um do tipo n foi escolhido uma vez que a sua condutividade é superior em comparação com a outra alternativa.

No que diz respeito ao comutador ligado a V_{in} , ao contrário dos demais casos até agora apresentados, ele traduz-se em dois *mosfets* em paralelo, ambos de tipos diferentes. A razão pela qual tal abordagem é necessária é o facto de que a tensão V_{in} , ao contrário dos casos anteriores, pode variar entre 0 V e 3.3 V.

No caso de V_{in} apresentar uma tensão de 0 V e o comutador apenas ser constituído por um *mosfet* do tipo p, independentemente da tensão que se aplica-se à *gate*, este nunca iria conduzir pois V_{GS} nunca seria menor que V_{th-p} . Pelo contrário, se o comutador apenas fosse constituído por um *mosfet* do tipo n, este iria conduzir uma vez que, se aplicados 3.3 V ao terminal *gate*, a tensão V_{GS} do *mosfet* seria maior que V_{th-n} .

Quando V_{in} apresenta a tensão mais elevada, o caso muda totalmente de figura pois desta feita é o *mosfet* do tipo n que nunca conduz ao passo que o *mosfet* do tipo p já consegue cumprir a condição $V_{GS} < V_{th-p}$, quando aplicada uma tensão de 0 V ao terminal *gate*.

Nos casos em que V_{in} apresenta valores intermédios, existe a possibilidade de ambos os *mosfets* entrarem em condução mas com condutividades mais baixas comparativamente aos casos extremos onde os *mosfets*, cada um no seu respetivo caso, apresentam as suas condutividades máxima e mínima. Deste modo, apesar de neste caso as condutividades dos *mosfets* não serem as mais elevadas, ambos estão em condução contribuindo assim para uma maior uniformização da condutividade do comutador ao longo de toda a excursão do sinal V_{in} .

Por último, existem também os comutadores que permitem não só descarregar os condensadores que interligam os quatro *arrays* de condensadores como também carregar os condensadores destes mesmos *arrays* com o valor da tensão a converter. Apesar de estes comutadores serem em tudo semelhante aos que permitem conectar os condensadores em paralelo à massa virtual do *DAC*, a sua constituição não passa apenas por um *mosfet* do tipo n mas sim de dois *mosfets* em paralelo, um do tipo p e outro do tipo n. Neste caso, a razão para tal configuração não é a excursão do sinal à entrada do comutador mas sim a compensação das capacidades parasíticas que são induzidas neste elemento.

Segundo as simulações, caso existisse apenas um *mosfet* do tipo n, ocorria uma discrepância nas tensões que o *DAC* gerava que mais tarde se viriam a refletir em parâmetros como o *INL* ou *DNL* inconvenientes. Deste modo, ao acrescentar um *mosfet* do tipo p, estas incongruências nas tensões geradas foram reduzidas.

Em suma, o resultado final no que respeita aos comutadores⁵, ao nível do *mosfet*, é apresentado na figura 4.11.

⁵ Informação adicional sobre os comutadores em Anexo IV – Banco de Condensadores ou *DAC*

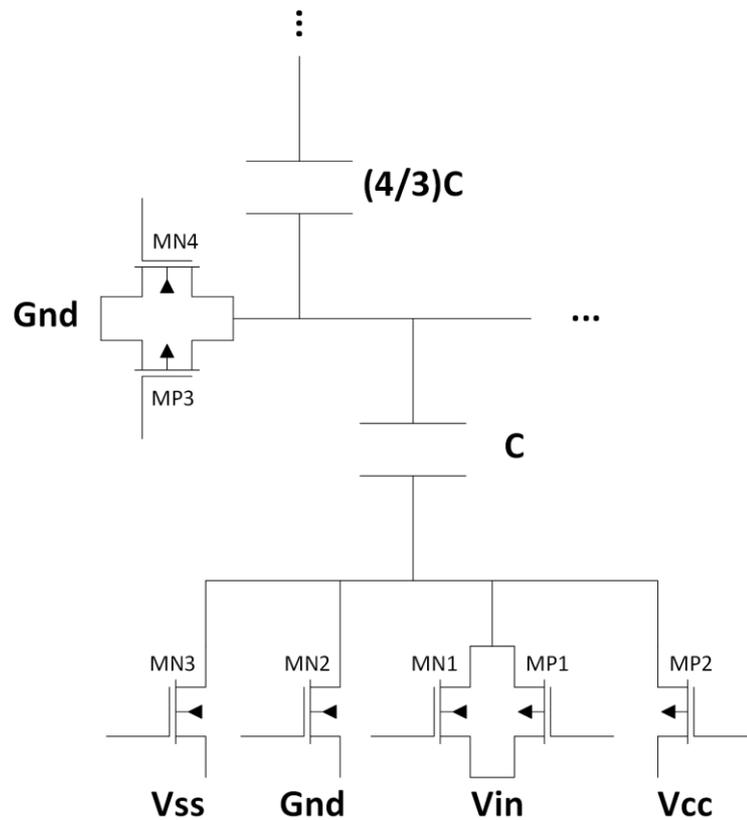


Figura 4.11 Configuração dos comutadores em baixo nível

Quanto aos sinais que se conectam com as *gates* dos *mosfets* presentes no *DAC*, eles têm origem na máquina de estados e representam o principal elo de ligação entre estes dois blocos.

4.3.3 Análise de frequências de entrada

De maneira a perceber qual o desempenho futuro de todo o sistema, um estudo sobre a capacidade que o banco de condensadores tem em reter o sinal de entrada quando este último apresenta frequências diferentes pode revelar-se bastante vantajoso. Deste modo, a melhor análise passa por recolher as tensões que o *DAC* consegue recolher e espelhar esses valores num espetro de frequências.

As figuras 4.12, 4.13 e 4.14 mostram os resultados obtidos para diferentes frequências de sinais de entrada.

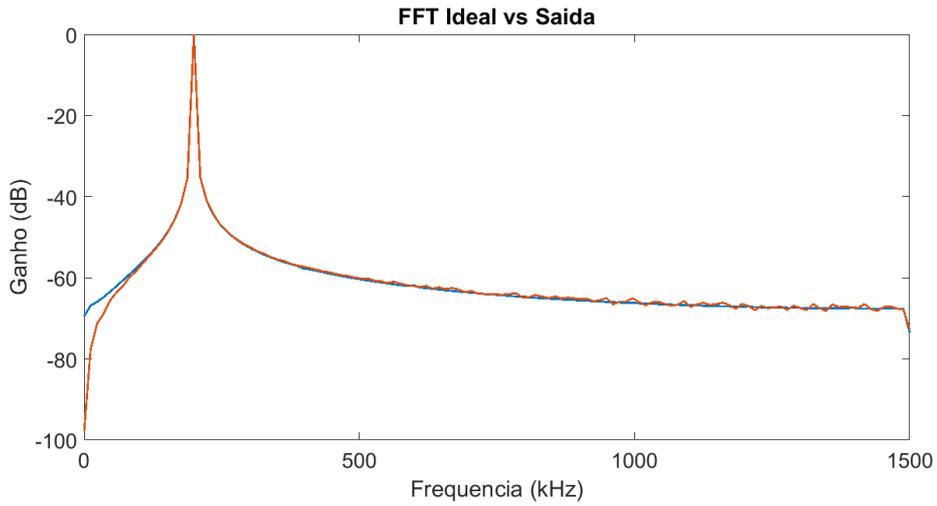


Figura 4.12 FFT de saída e ideal com frequência de entrada de 164 kHz

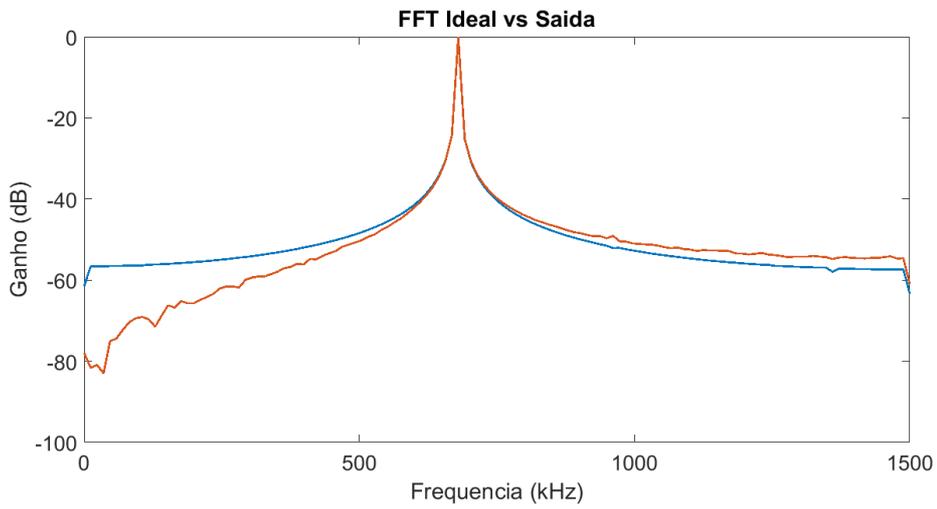


Figura 4.13 FFT de saída e ideal com frequência de entrada de 726 kHz

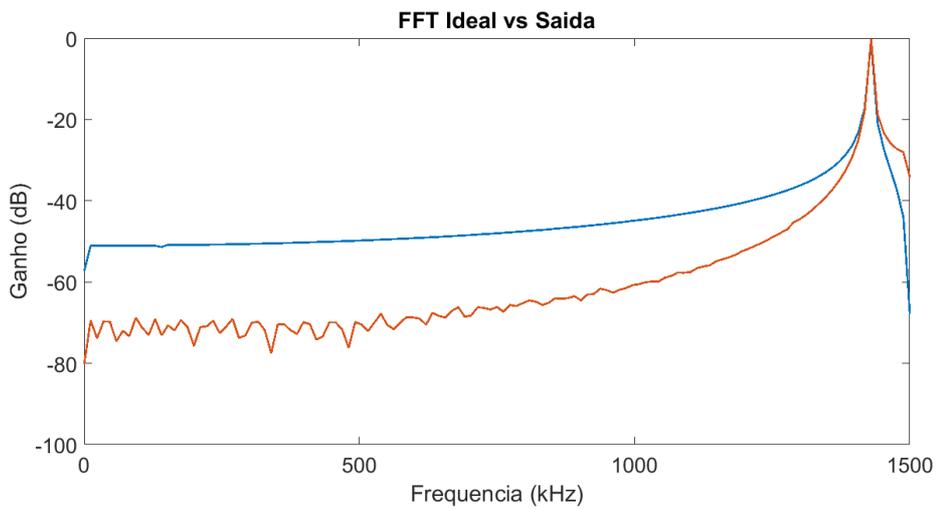


Figura 4.14 FFT de saída e ideal com frequência de entrada de 1.43 MHz

Tal como mostram as figuras, à medida que a frequência do sinal de entrada vai aumentando, o sinal que se encontra nos condensadores vai perdendo qualidade (representado a azul) face ao sinal ideal (representado a vermelho).

Se analisarmos a equação que rege o comportamento dos condensadores, $i_C = C * \frac{dv}{dt}$, verifica-se que quanto maior é a derivada da tensão exercida sobre um condensador, maior é a corrente a que ele está submetido. Neste sentido, quanto maior for a frequência do sinal de entrada maior será a corrente a que os condensadores do *DAC* estão sujeitos. Relembrando aquele que é a constituição do *DAC*, quando o *ADC* está na fase de *Sample*, os comutadores que se conectam aos condensadores em série estão ligados para que a tensão a converter possa ser armazenada. Ora, se a frequência do sinal de entrada tiver um valor considerável, a corrente a que esses comutadores estão sujeitos também tem valores relativamente altos, situação esta que obrigatoriamente gera uma tensão aos terminais do comutador. Devido a isto, os condensadores não conseguem ficar carregados com a tensão que idealmente deveriam ter pois existe sempre uma tensão residual aos terminais do comutador.

No cômputo geral, o desempenho é satisfatório uma vez que para frequências até cerca de 700 kHz as diferenças não apresentam valores elevados. Já para frequências acima desse valor até à frequência máxima admissível para este *ADC* (1.5 MHz), o desempenho é ligeiramente inferior face a frequências mais baixas.

4.4 Máquina de estados

A máquina de estados pode ser considerada o cérebro deste sistema uma vez que é este elemento que dita o que acontece e quando acontece. No que diz respeito à interação Homem-*ADC*, toda e qualquer ordem realizada por parte do utilizador é feita apenas através da máquina de estados que, posteriormente, reagirá em conformidade dando a conhecer a todos os outros elementos qual a função que devem desempenhar.

Por outro lado, quando o *ADC* pretende notificar o utilizador para o final de uma conversão por exemplo, esta é consumada através de um *bit*, o *EOC*, que é também providenciado pela máquina de estados. De um modo geral, todos os blocos à exceção da máquina de estados são exclusivamente manipulados pelo *ADC* em si, não havendo desta forma qualquer tipo de interação com o exterior.

4.4.1 Conceção do circuito

Depois de concebidos e estudados todos os constituintes do *ADC* passa-se então à fase de conceção da máquina de estados⁶ que, devido à versatilidade do *software Cadence*, se traduz essencialmente através de um ficheiro de código *Verilog*. Deste modo, todo o processo de criação e otimização do circuito eletrónico é realizado pelo *software*, sendo apenas o seu comportamento manipulado pelo utilizador através desse mesmo ficheiro de código.

Outro aspeto importante que o utilizador tem que manipular são as denominadas *timing constraints* que são essencialmente expressões que definem quais os períodos de tempo máximos e mínimos em que determinadas entradas e saídas do bloco podem mudar de estado. Com o auxílio das *timing constraints* o *software* consegue realizar as otimizações e mudanças necessárias ao circuito para que todos esses tempos possam ser respeitados.

No caso de existir alguma violação destes tempos, o utilizador é notificado e torna-se necessário fazer uma revisão dos tempos impostos à máquina de estados. Caso contrário, é garantido ao utilizador que o circuito gerado é capaz de traduzir o comportamento imposto pelo ficheiro de código *Verilog* nos tempos requeridos pelas *timing constraints*.

4.4.2 Comportamento da máquina de estados

Tal como já mencionado em secções anteriores, a máquina de estados que comanda este *ADC*, para conseguir realizar uma conversão, tem que percorrer no total 15 estados que representam as fases de *Sample*, *Hold*, as doze fases que permitem descobrir os doze *bits* do *ADC* e, por último, uma fase para a calibração do comparador. No entanto, esta máquina de estados, para além dos 15 estados já mencionados, apresenta mais um que é em tudo semelhante ao estado de calibração do comparador em modo de conversão contínua.

A principal razão para a inclusão de mais um estado foi o facto de o comparador, no momento em que se liga o *ADC* através do *bit Enable*, ter uma grande probabilidade de estar completamente descompensado de maneira a que o seu *offset* apresente valores inapropriados. Deste modo, para que se proceda à redução do *offset*, este estado adicional é acionado ciclicamente até que a saída do comparador mude de estado. A título de exemplo, assumase um *offset* positivo que culmina num *output* de nível lógico 1 por parte do comparador. No caso de o *offset* apresentar um valor elevado, durante um determinado período de tempo o seu *output* permanece inalterado, apesar das sucessivas calibrações. Quando o *offset* é reduzido até ao valor

⁶ Informações adicionais sobre a máquina de estados em Anexo V – Máquina de estados

mínimo, o *output* do comparador muda desta feita para o nível lógico 0, informando assim a máquina de estados que o comparador está calibrado. De seguida, o utilizador é notificado através do *bit Ready* que o *ADC* está apto a realizar conversões.

Depois de todo este processo, o utilizador necessita de ativar o *bit Start_conv* para que uma conversão possa ser consumada. No caso de este *bit* estar permanentemente a 1, são realizadas conversões de forma contínua à taxa máxima de 3 *MSPS*. Se o utilizador pretender realizar conversões a uma frequência mais baixa, basta gerar picos a 1 do *bit Start_conv* tão espaçados no tempo quanto o período correspondente à frequência a que pretende amostrar o sinal analógico de entrada.

É importante referir que, caso seja pretendida apenas uma conversão, o tempo máximo que o *bit Start_conv* deve permanecer a um 1 é aproximadamente $333\text{ ns} \left(\frac{1}{3\text{ MSPS}}\right)$, uma vez que a máquina de estados somente analisa este *bit* no início de cada conversão, ou seja, na fase de *Sample*. Por isso, se ele permanecer a 1 durante 350 *ns* por exemplo, duas conversões são realizadas uma vez que ao final de 333 *ns* o *bit* ainda permanece a 1. Apesar de este vir a mudar de estado cerca de 17 *ns* depois, isso em nada interfere com o funcionamento da máquina de estados uma vez que este *bit* não apresenta qualquer relação com os estados intermédios.

Quando a conversão se encontra nos estados referentes aos vários *bits*, a comparação é iniciada quando o sinal *clock* está na sua fase ascendente, sendo verificado o seu resultado na transição descendente do *clock* seguinte. Deste modo, o resultado da comparação tem um tempo máximo de 11 *ns* para estabilizar. Ainda na fase descendente do *clock*, a máquina de estados altera o estado dos comutadores presentes no *DAC* consoante o resultado da comparação imediatamente anterior.

No final de cada uma das conversões, o *bit EOC* muda para o nível lógico 1 e informa o utilizador que uma conversão foi realizada. Quanto ao resultado da conversão, ela encontra-se no barramento de 12 *bits ADC_Out*.

Por último, para que tudo funcione nos tempos devidos, na entrada *Clk* é aplicado o sinal de *clock* com uma frequência de 45 *MHz*, permitindo assim à máquina de estados realizar conversões a uma taxa máxima de 3 *MSPS*.

Todos os sinais até agora mencionados que representam tanto entradas como saídas deste bloco são aqueles que podem ser manipulados ou consultados pelo utilizador. É importante salientar que, para além destes sinais, a máquina de estados apresenta um interface ainda mais alargado. A título de exemplo, visto que é este bloco que comanda os comutadores presentes no *DAC*, a máquina de estados tem obrigatoriamente que contemplar no seu leque de *outputs* alguns reservados para os controlar. Outros exemplos podem ser as saídas positiva e negativa

do comparador que são entradas do ponto de vista da máquina de estados ou *outputs* desta última que informam o comparador quando é que uma calibração deve ser efetuada.

A figura 4.15 mostra todo o interface da máquina de estados.



Figura 4.15 Interface da máquina de estados

4.5 Circuito final

Como etapa final, a melhor forma de perceber se todos os blocos têm a capacidade de funcionar em conjunto passa por realizar as respetivas ligações entre os vários elementos e submeter o sistema resultante a situações com que se poderá deparar quando implementado em silício.

A figura 4.16 mostra os sinais mais importantes durante uma das conversões realizadas.

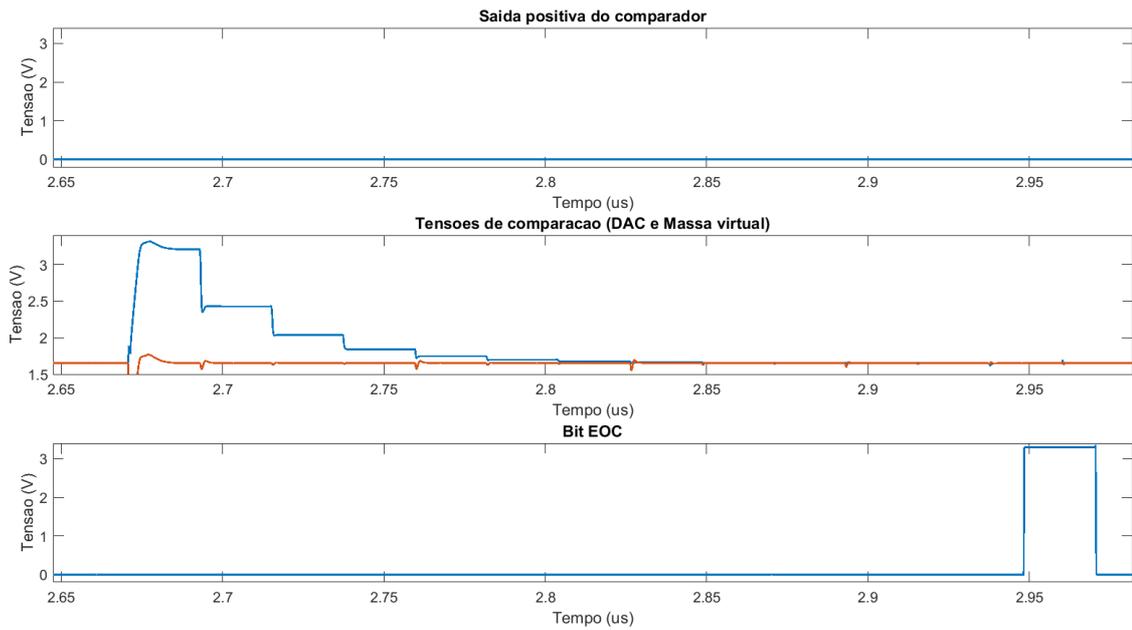


Figura 4.16 Sinais durante uma conversão

Os sinais apresentados na figura 5.16 mostram qual é a sequência que os vários sinais apresentam quando a tensão a converter apresenta o valor mais baixo possível, ou seja, 0 V. Tal se pode concluir vislumbrando a saída do comparador que apresenta o valor zero durante todos os estados referentes aos 12 *bits* que o *ADC* apresenta. Deste modo, quando o *bit EOC* transitar para o nível lógico 1 (último gráfico), o barramento de saída será preenchido com 12 *bits* com o nível lógico 0.

É também importante referir que várias conversões foram consumadas, não espelhando qualquer uma delas qualquer tipo de problemas relativamente ao sistema final.

5. FASE DE *LAYOUT*

Quando o objetivo de algum projeto é conceber um *chip*, o processo de criação segue uma metodologia sequencial. Posto isto, logo depois de todo o circuito estar dimensionado e simulado, segue-se a fase onde esse mesmo circuito é transferido para uma *wafers* de silício. Nesta fase, com o auxílio das máscaras que o processo de fabrico fornece, esta transferência pode ser consumada. Quando dispostas da forma correta, estas máscaras permitem criar elementos como *mosfets* tanto do tipo p ou n, resistências e condensadores por exemplo.

No entanto, devido aos elevados custos de fabrico, é importante conseguir racionar toda a área de silício da melhor forma de maneira a que seja reduzida ao máximo a área inutilizada. Para a maioria das pessoas, uma décimas de milímetro são insignificantes mas, a esta escala, quando um circuito eletrónico tem essas décimas de milímetro a mais e é produzido em massa, isso pode traduzir-se num aumento dos custos bastante significativo.

Contudo, com esta redução da área ocupada, todos os elementos que compõe o circuito em questão ficam extremamente aproximados uns em relação aos outros. Este fator pode revelar-se problemático uma vez que a indução de alguns comportamentos indesejados no circuito podem revelar-se uma realidade, podendo até mesmo interferir com a integridade do funcionamento do mesmo.

Apesar de parecer fácil a transcrição de um circuito eletrónico para uma *wafers* de silício, uma vez que todas as medidas são conhecidas, este processo não se revela assim tão simples devido à ponderação que é preciso ser feita entre a área ocupada e o funcionamento final do sistema.

No decorrer deste capítulo, serão abordados todos os blocos constituintes do *ADC*, com principal foco nos seus respetivos *layouts*.

5.1 Comparador

Visto que este bloco desempenha um papel vital neste sistema e apresenta requisitos bastante apertados, torna-se necessário adotar um cuidado especial no desenho deste elemento. Deste modo, visto que o circuito que compõe o comparador apresenta dois ramos idênticos, escolheu-se um desenho o mais simétrico possível para que o comportamento do circuito seja também ele o mais semelhante possível em ambos os ramos. Outra técnica utilizada que merece

destaque é o facto de que tanto os *mosfets* do tipo p como do tipo n estão rodeados por anéis do tipo n e do tipo p, respetivamente. Tal como referido em [3], esta prática melhora a performance do circuito quando sujeito a algumas situações adversas, nomeadamente quando exposto a radiação forte.

Posto isto, o *layout* resultante do comparador⁷ é apresentado na figura 5.1.

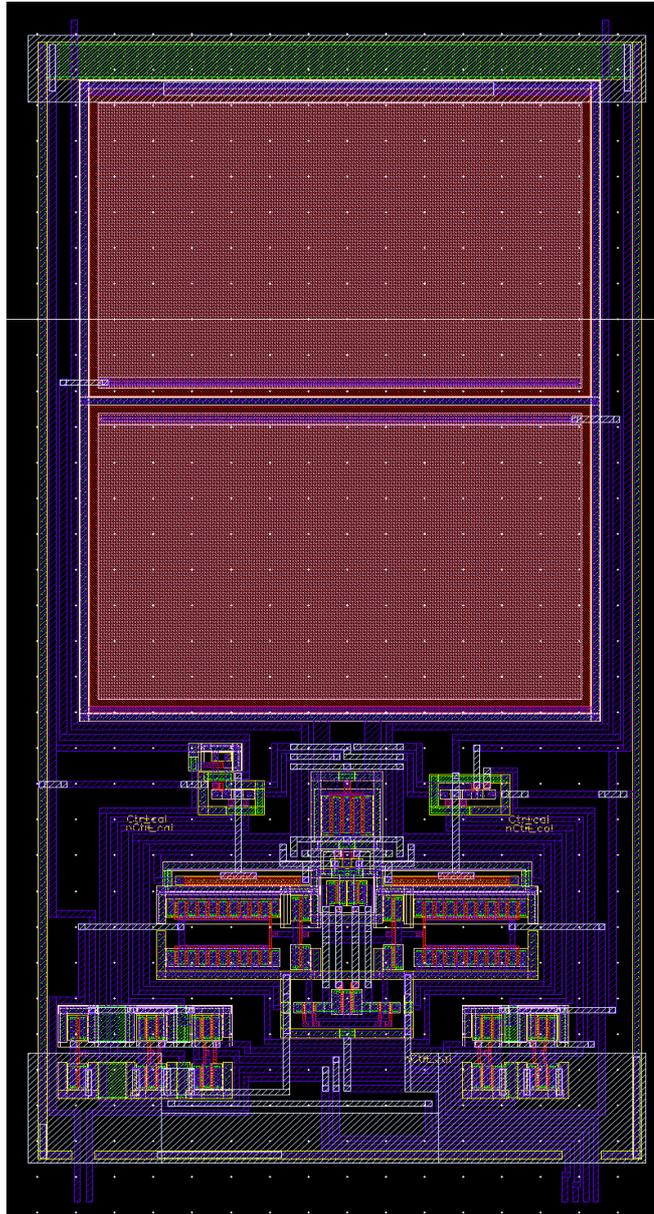


Figura 5.1 Layout do comparador

⁷ Informação adicional sobre o *layout* do comparador em Anexo I - Comparador

5.2 Amplificador operacional

Outro dos elementos que é parte integrante deste *ADC* é um amplificador operacional que, quando configurado como seguidor de tensão, providência a massa virtual do *DAC*. Para além da técnica utilizada no comparador e referida em [3], ambos os *mosfets* que integram os respetivos ramos do andar diferencial foram “encaixados” um no outro, configuração esta muitas vezes denominada *combed fingers*.

A figura 5.2 seguinte mostra o *layout* concebido para o amplificador operacional⁸.

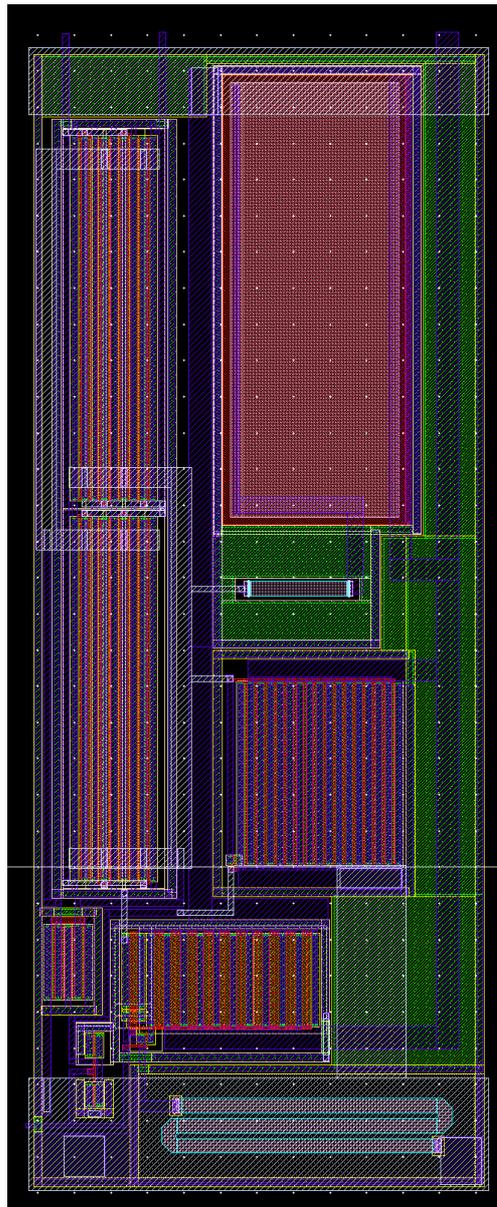


Figura 5.2 Layout do amplificador operacional

⁸ Informação adicional sobre o *layout* do amplificador operacional em Anexo III – Amplificador operacional

5.3 Banco de condensadores ou DAC

No que ao banco de condensadores diz respeito, a escolha passou por criar uma matriz de condensadores para uma melhor acomodação dos mesmos. Ainda na parte inferior desta matriz, foram introduzidos os comutadores que controlam os condensadores para uma posterior melhor interação entre a máquina de estados e o DAC. Quanto a técnicas de *layout*, apenas se contemplou a abordada em [3]. A figura 5.3 mostra o *layout* final do DAC⁹.

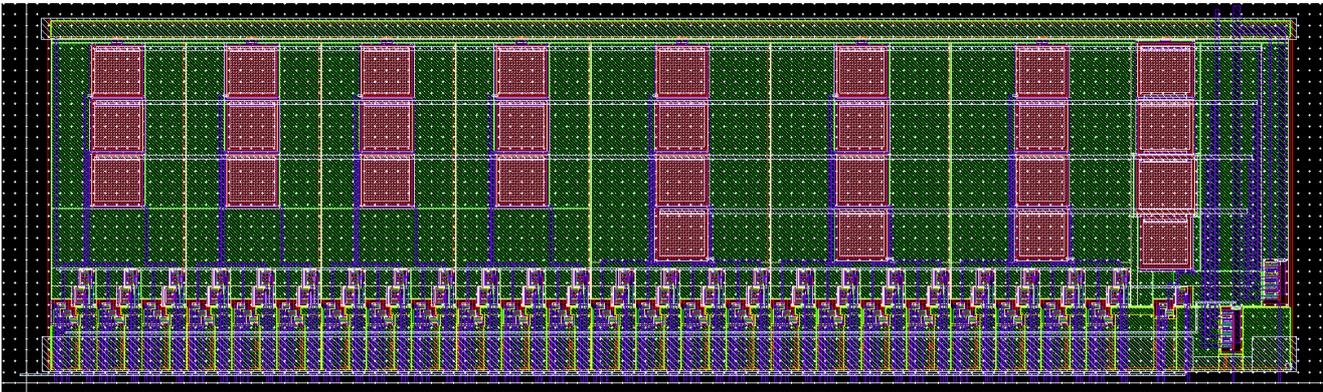


Figura 5.3 Layout do banco de condensadores

5.4 Máquina de estados

De modo a que todo o ADC fica-se representado sob a forma de *layout*, procedeu-se à criação do correspondente à máquina de estados. Tal como na fase de conceção do circuito, é o *software* que, amparado por toda a informação produzida nessa fase, cria todo o *layout* do sistema em questão.

Durante todo este processo, o *software* acrescenta todos os elementos necessários para um bom funcionamento da máquina de estados bem como uma *clock tree*. Este último elemento consiste, tal como o nome indica, numa árvore que se ramifica por todo o circuito com o objetivo de fazer com que o sinal de relógio consiga atingir todos os componentes que dele necessitam. Como tal, todas estas derivações do sinal de *clock* provocam atrasos na propagação do mesmo, situação esta que pode colocar em questão as *timing constraints* impostas aquando

⁹ Informação adicional sobre o *layout* do DAC em Anexo IV – Banco de condensadores ou DAC

da fase de conceção do circuito. Devido a esta possibilidade, são gerados relatórios à medida que o processo se vai desenrolando que ditam o cumprimento ou não de todos esses tempos. Em caso negativo, torna-se necessário verificar qual ou quais *constraints* são as transgredidas de maneira a que uma nova revisão nesses mesmos tempos possa ser feita de uma forma mais direta e objetiva.

Outro aspeto que o utilizador deve contemplar é o posicionamento de todas as entradas e saídas do sistema. De um modo geral, todos os blocos que são criados em *layout* não tem como objetivo principal funcionar por si próprio mas sim fazer parte de um sistema maior, composto por vários blocos diferentes. Como tal, existe a necessidade de todos estes elementos comunicarem entre si para que o circuito final apresente o comportamento desejado. Deste modo, é gerado um ficheiro com o posicionamento de todos os *inputs* e *outputs* da máquina de estados e, caso algum destes sinais não se encontre no local mais adequado, o utilizador tem a possibilidade de o reposicionar. Quando tal é necessário, o processo tem que ser repetido novamente.

Devido à facilidade em reposicionar todos os sinais de entrada e saída do circuito, em certas situações os relatórios gerados nesta fase podem revelar tempos de propagação dos sinais diferentes, revelando violações de *constraints* que em iterações anteriores eram respeitadas. Este fator também merece atenção pois o posicionamento errado de determinados sinais pode hipotecar a integridade do circuito final, apesar da grande maioria destes portos de entrada e saída não ter influencia para tal. Regra geral, são sinais como o de *Clk* por exemplo que merecem mais atenção, especialmente quando os circuitos gerados apresentam dimensões consideráveis. Nestes casos, a opção a tomar deve ser a de colocar este sinal de entrada numa posição média em relação a todo o circuito para que ele possa chegar da forma o mais uniforme e rápida possível a todos os pontos necessários.

Quando todo este processo é concluído e todas as *timing constraints* e posicionamentos finais são respeitados, o *layout* resultante representa uma máquina de estados com o comportamento indicado no ficheiro de código *Verilog* inicialmente criado, com todos os tempos impostos nas *timing constraints* respeitados.

A figura 5.4 apresenta o aspeto final da máquina de estados¹⁰.

¹⁰ Informação adicional do *layout* da máquina de estados em Anexo V – Máquina de estados

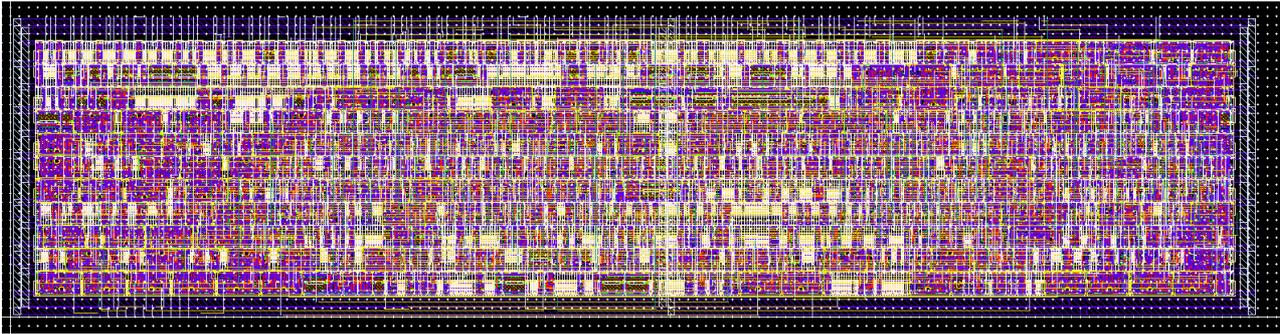


Figura 5.4 Layout da máquina de estados

5.5 Circuito final

Depois de concebidos os respetivos *layouts* de todos os blocos constituintes do *ADC*, torna-se então possível interligar todos eles de maneira a criar o circuito final. Depois de interligados todos os circuitos e verificadas todas as ligações entre eles, o circuito final apresenta umas dimensões aproximadas de $750 \mu\text{m}$ de comprimento por $350 \mu\text{m}$ de largura, perfazendo uma área total de 0.2625 mm^2 .

A figura 5.5 apresenta o *layout* do circuito final.

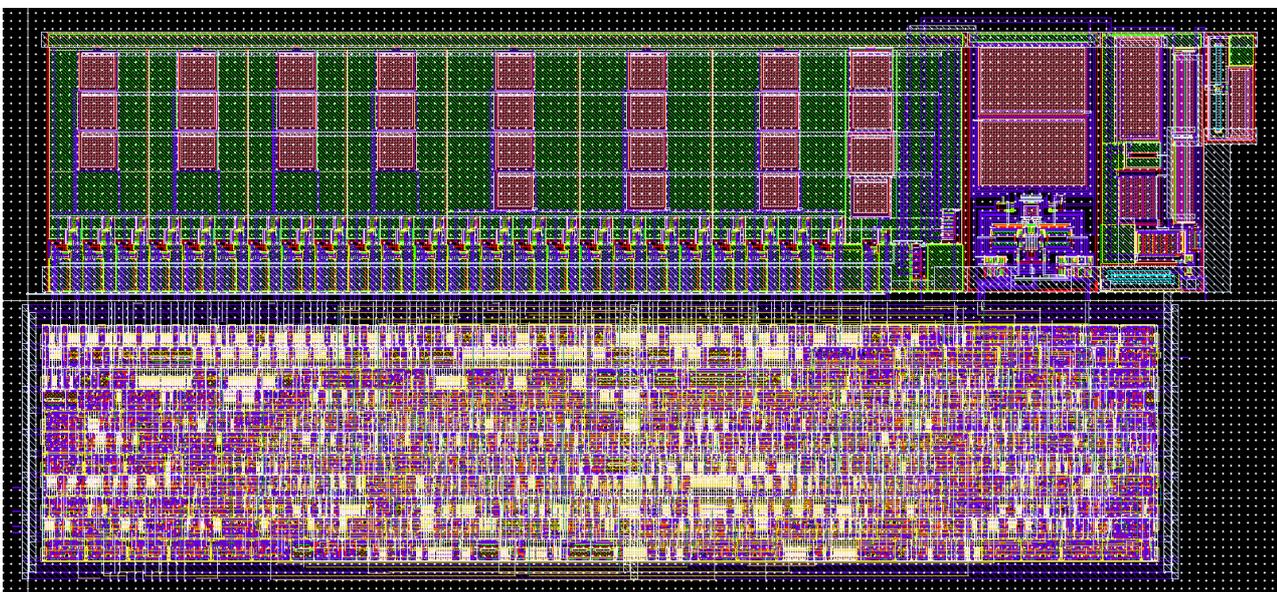


Figura 5.5 Layout do circuito final

6. SIMULAÇÃO PÓS-*LAYOUT*

Quando um determinado circuito é convertido no seu respectivo *layout*, o seu comportamento, quando estiver sob a forma de um *chip*, não será exatamente igual ao comportamento que o circuito apresentava aquando da fase de simulação de baixo nível. Tal como anteriormente referido, a proximidade entre os vários elementos pode gerar comportamentos indesejados que devem ser avaliados de maneira a perceber qual a sua influência no resultado final.

Como tal, com o auxílio do *software* utilizado, o *Cadence*, procedeu-se à extração de todos os circuitos finais através dos *layouts* criados, circuitos estes que para além dos elementos incluídos nos mesmos na fase de simulação de baixo nível também contemplam os elementos induzidos na fase de *layout* como capacidades ou impedâncias parasitas. Deste modo, realizaram-se novas simulações por forma a perceber se o comportamento de todos os circuitos foi comprometido ou se pelo contrário esses mesmos comportamentos não apresentam razões que os invalidem.

6.1 Comparador

Neste elemento em particular, as circunstâncias que merecem maior atenção são o *offset* de entrada do comparador bem como o tempo que a saída deste bloco necessita para estabilizar, tudo isto simulado para todos os *process corners* às temperaturas de -30°C , 30°C e 130°C .

6.1.1 Tempo de estabilização do *output*

Tal como mencionado na fase de simulação de baixo nível, o comparador apenas dispõe de cerca de 11 *ns* para ter a sua saída com um valor credível de maneira a não hipotecar o funcionamento do circuito. Nesse sentido, a figura 6.1 revela os resultados conseguidos durante as simulações realizadas.

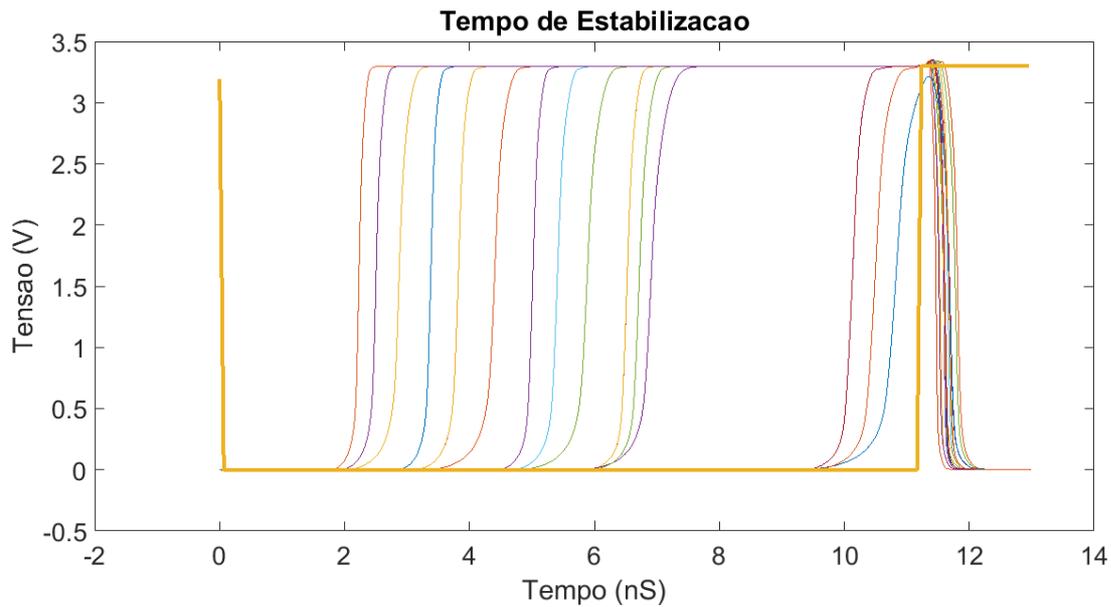


Figura 6.1 Tempo de estabilização do output do comparador (pós-layout)

Tal como mostra a figura 6.1, no momento em que o sinal de *clock* está em fase ascendente, em todas as situações o output está no nível lógico 1. Apesar de numa situação em específico o *output* não se apresentar totalmente estável na transição do *clock*, é uma situação que não apresenta grande problema uma vez que é notório o nível lógico que o sinal pretende apresentar.

6.1.2 Análise de *offset*

Relembrando a fase de simulação de baixo nível, o *offset* de entrada máximo rondava os $200\ \mu V$ quando o máximo estipulado nessa fase tinha o valor de $400\ \mu V$. Apesar de no pós-*layout* existir uma grande possibilidade de deterioração das características de qualquer circuito, o valor máximo desejado permanece nos $400\ \mu V$. A figura 6.2 mostra os resultados conseguidos nesta fase.

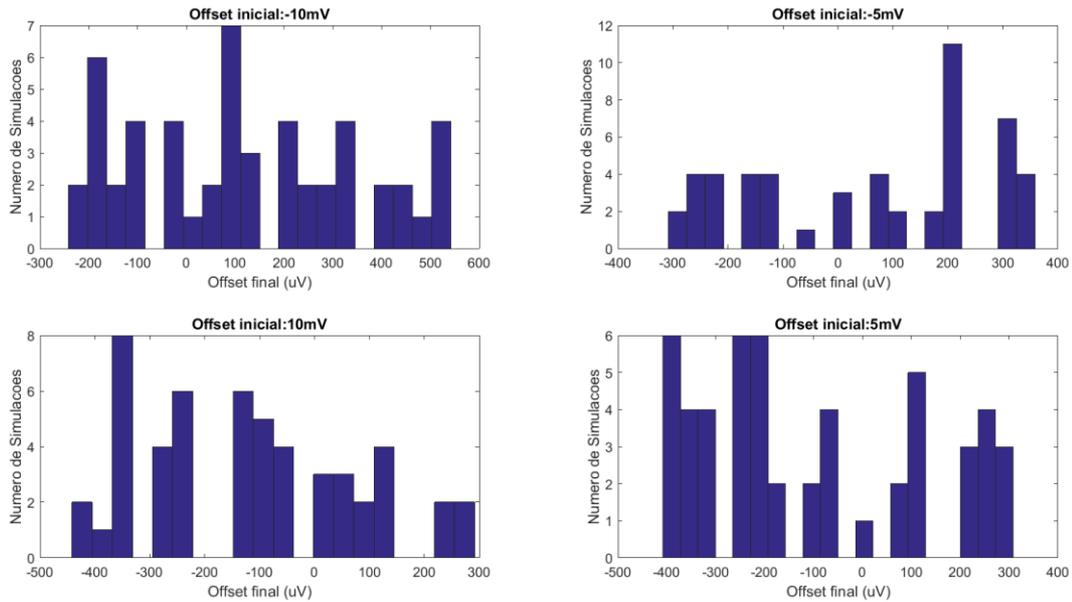


Figura 6.2 Offset de entrada do comparador

Tal como mostra a figura 6.2, a esmagadora maioria dos casos apresenta um *offset* de entrada menor que o estipulado. No entanto, em certas circunstâncias, este valor ultrapassa ligeiramente os $400 \mu V$.

No cômputo geral, em relação ao *offset* de entrada os valores são aceitáveis uma vez que as situações que infringem os limites impostos não o fazem de uma forma demasiadamente exagerada.

6.2 Amplificador operacional

No que ao amplificador operacional configurado como seguidor de tensão diz respeito, a situação que se poderá revelar mais problemática é sem dúvida a diferença da tensão de saída para a tensão ideal, neste caso $1.65 V$.

A figura 6.3 revela os dados adquiridos nas simulações realizadas.

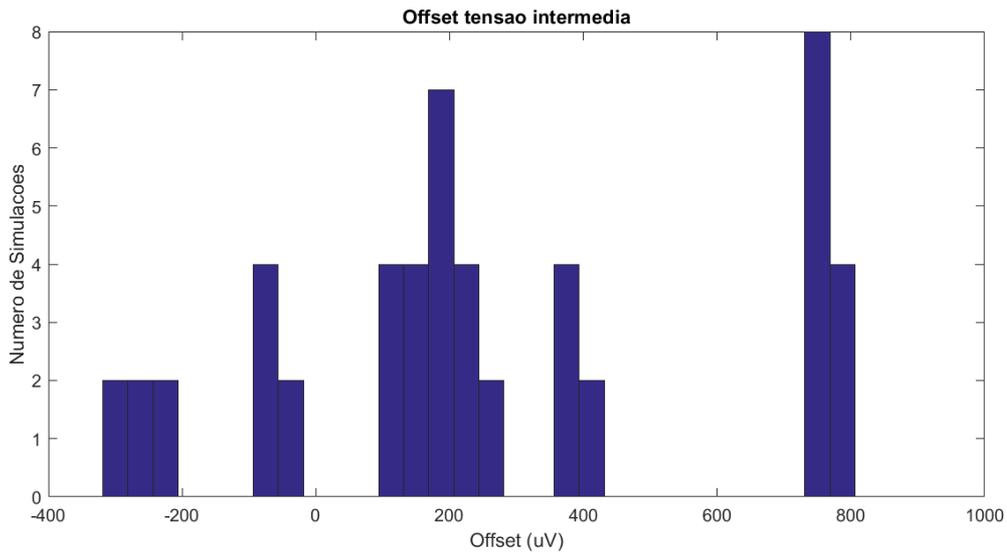


Figura 6.3 Offset de saída do amplificador operacional (pós-layout)

Tal como mostra a figura 6.3, os resultados não apresentam diferenças significativas em relação aos valores conseguidos nas simulações de baixo nível. Deste modo, o resultado final para este elemento é bastante satisfatório.

6.3 Banco de condensadores ou DAC

Para este bloco em particular, as simulações de pós-layout que se poderão revelar mais produtivas são as referentes à capacidade de retenção do sinal de entrada por parte do DAC. Tal como na fase de baixo nível, simulações com diferentes frequências de sinal de entrada foram realizadas, estando os seus resultados apresentados nas figuras 6.4, 6.5 e 6.6.

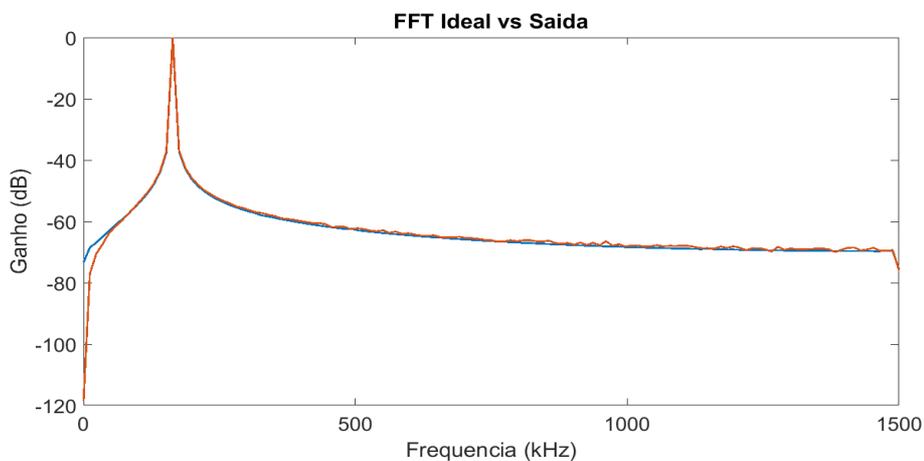


Figura 6.4 FFT de saída e ideal com frequência de entrada de 164 kHz (pós-layout)

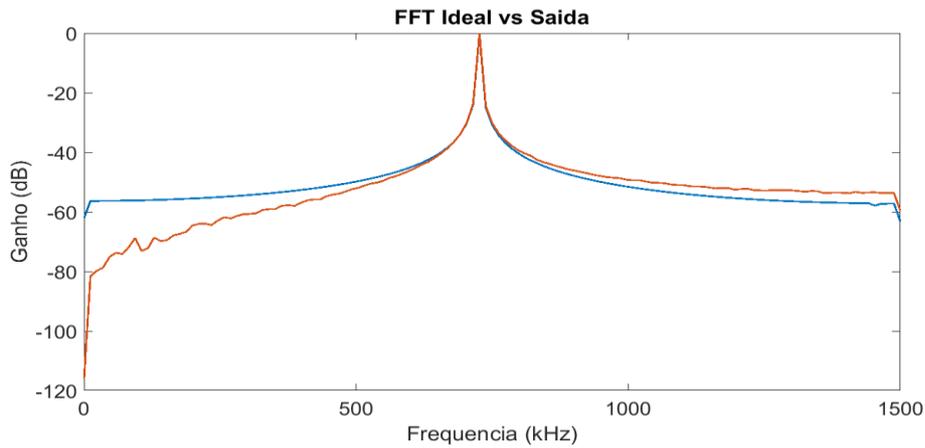


Figura 6.5 FFT de saída e ideal com frequência de entrada de 726 kHz (pós-layout)

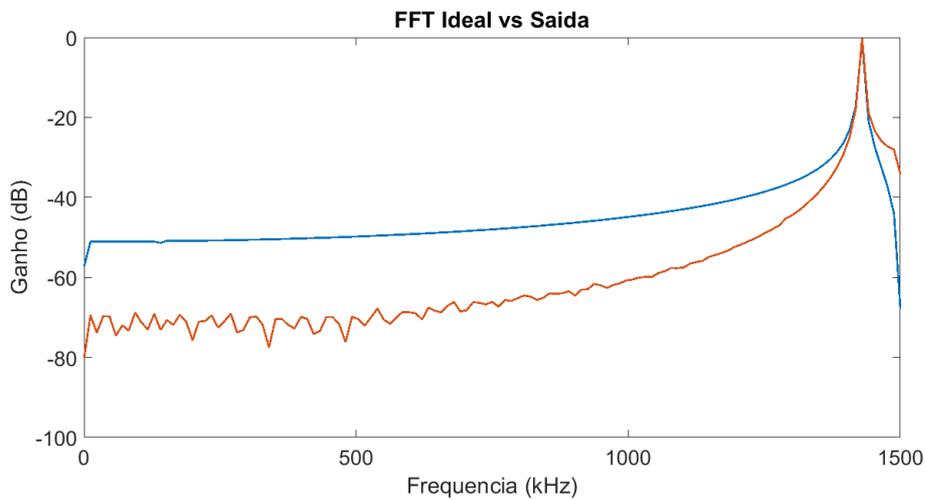


Figura 6.6 FFT de saída e ideal com frequência de entrada de 1.43 MHz (pós-layout)

Tal como mostram os resultados anteriores, as diferenças relativamente as simulações de baixo nível são diminutas, o que indica que não foi comprometido o funcionamento final do DAC.

6.4 Máquina de estados

Relativamente a este bloco, visto que é neste elemento onde toda a dinâmica de funcionamento é imposta, a melhor forma de perceber se tudo funciona conforme o previsto é simular a totalidade do circuito. Nesse sentido, o funcionamento da máquina de estados revelar-se-á aprovado caso o circuito final não apresente qualquer tipo de problema.

6.5 Circuito final

Depois de verificados e validados os comportamentos de todos os blocos constituintes do *ADC*, a última etapa passa pela simulação de todo o circuito, procedendo assim a todas as ligações necessárias entre os vários elementos constituintes do sistema final.

De seguida apresentam-se na figura 6.7 os sinais mais importantes no decorrer de uma conversão.

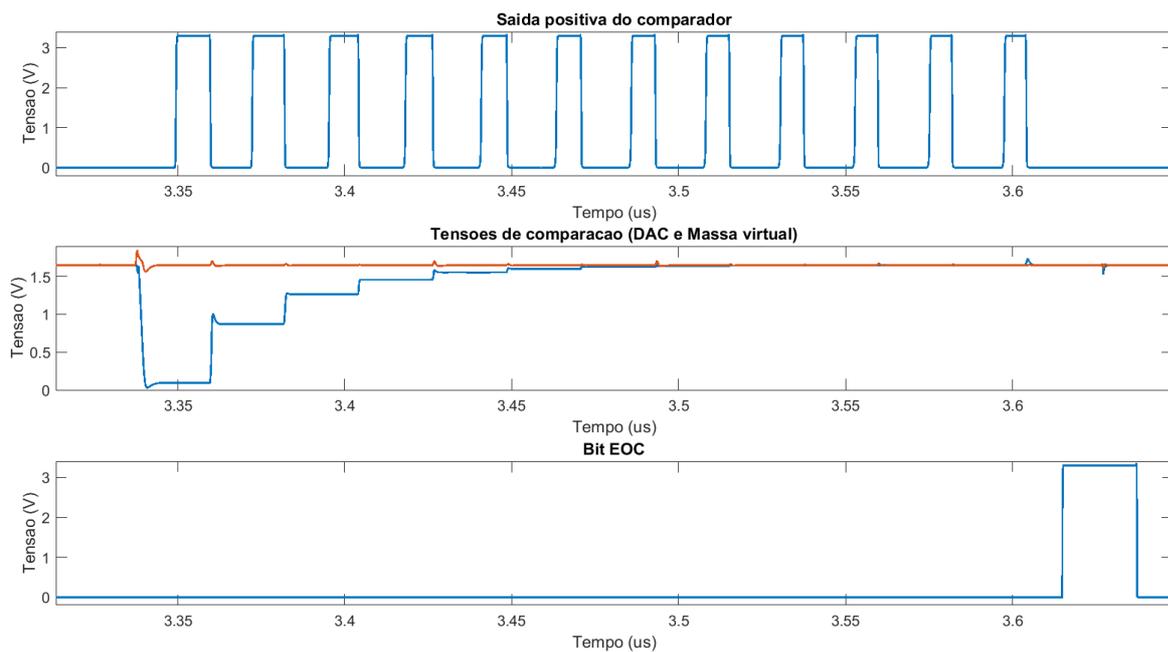


Figura 6.7 Sinais durante uma simulação (pós-layout)

Durante a simulação apresentada na figura 6.7, a tensão que se pretendia converter tinha o valor mais alto passível de ser convertido, neste caso, 3.3 V. Tal poderá ser distinguido pelo facto de que a saída do comparador, em todos os estados referentes aos 12 *bits*, apresentar à sua saída o nível lógico 1. Deste modo, quando o *bit EOC* transitar para o nível lógico 1 (último gráfico), o barramento de saída será preenchido com 12 *bits* com o nível lógico 1.

Apesar de várias conversões terem sido consumadas durante a simulação realizada, por razões de percetibilidade apenas uma foi mostrada. No decorrer da mesma simulação, tal como a conversão aqui apresentada, todas as outras se concretizaram sem qualquer espécie de violação. Posto isto, pode dizer-se que todos os circuitos constituintes do *ADC* funcionam entre si, resultando desta forma um sistema válido e de possível implementação em silício.

7. CONCLUSÕES

Depois de analisada qual a finalidade do *ADC* e em que circunstâncias o sistema iria ser integrado, seguiu-se um estudo de maneira a perceber quais as diferentes possibilidades que existem atualmente. Desta forma, vários artigos científicos especializados e publicações literárias foram consultadas para que a idealização do sistema final começasse a ser construída da forma mais clara e consistente possível.

Quando as especificações do sistema final ficaram definidas, toda a informação recolhida até então foi sujeita a uma seriação pois várias das alternativas ponderadas não apresentavam qualquer tipo de vantagem face as circunstâncias. De entre todas as possibilidades, a arquitetura que pareceu mais adequada foi sem dúvida a arquitetura *SAR* devido ao facto de esta apresentar algumas vantagens como baixos consumos elétricos, um princípio de funcionamento bastante simples e estar perfeitamente enquadrada na frequência de amostragem pretendida. Como tal, procedeu-se a um estudo mais minucioso das partes que constituem um *SAR ADC*: banco de condensadores, comparador e máquina de estados. Depois de estudados todos estes elementos, simulações foram realizadas para cada um deles de maneira a perceber se os seus respectivos usos não apresentavam qualquer tipo de entrave ao bom funcionamento do circuito final.

Concluída esta fase, todos os elementos foram reunidos para que o circuito final fosse submetido a um teste final de modo a perceber se tudo funcionava como o previsto. Todas as simulações não apresentaram qualquer tipo de problema, validando assim o circuito conseguido.

De seguida, todos os elementos foram submetidos à fase de *layout*. Depois de conseguidos e validados todos os *layouts* referentes aos vários blocos, novas simulações foram realizadas de maneira a perceber se algum efeito adverso resultante desta fase tinha colocado em causa o funcionamento de cada deles. Depois de validados todos os subsistemas, estes foram reunidos de modo a criar o *ADC* final para posteriores simulações. Os resultados conseguidos nesta fase mostraram que o circuito final funcionava tal como o pretendido, resultando assim numa validação deste para posterior implementação em silício.

7.1 Trabalho futuro

No que diz respeito a trabalho, são fundamentalmente duas as etapas que necessitam de ser concluídas. A primeira passa por materializar o sistema final sob a forma de um *chip*. Para finalizar, o *chip* deve ser submetido a uma bateria de testes para que se possa proceder à sua caracterização final (*DNL*, *INL*, *ENOB*, *SFDR* e *SNDR*).

7.2 Melhorias

Apesar de todas as simulações revelarem que o circuito final é funcional, existem vários aspetos que podem ser melhorados. Com isto, o *ADC* final pode vir a revelar melhorias em alguns dos seus atributos.

Desde logo, um estudo intensivo do *DAC*, e em particular dos seus comutadores, certamente que apresentará melhorias sobretudo da capacidade de retenção de sinais de frequências altas, traduzindo-se dessa forma numa performance bastante mais interessante do sistema final.

Outro aspeto que poderia ser melhorado é o *offset* do comparador. Tal como mencionado no decorrer deste documento, o *offset* da fase de baixo nível para a de pós-*layout* revelou um aumento bastante significativo. Isto poderá ser resolvido talvez com técnicas de *layout* mais apuradas ou poderá passar mesmo por um redimensionamento do circuito do comparador.

Por último, a dinâmica imposta pela máquina de estados poderá ser alterada de maneira a poupar alguma energia sobretudo quando o utilizador não pretende realizar conversões à taxa máxima que este *ADC* permite.

BIBLIOGRAFIA

- [1] S. Tao and A. Rusu, "A Power-Efficient Continuous-Time Incremental Sigma-Delta ADC for Neural Recording Systems," *IEEE Transactions on Circuits and Systems I-Regular Papers*, vol. 62, pp. 1489-1498, Jun 2015.
- [2] C.-Y. Chen, J. Wu, J.-J. Hung, T. Li, W. Liu, and W.-T. Shih, "A 12-Bit 3 GS/s Pipeline ADC With 0.4 mm² and 500 mW in 40 nm Digital CMOS," *IEEE Journal of Solid-State Circuits*, vol. 47, pp. 1013-1021, Apr 2012.
- [3] W. Liu, T. Wei, B. Li, P. Guo, and Y. Hu, "Design of a 12-bit 1 MS/s SAR-ADC for front-end readout of 32-channel CZT detector imaging system," *Nuclear Instruments & Methods in Physics Research Section a-Accelerators Spectrometers Detectors and Associated Equipment*, vol. 786, pp. 155-163, Jun 21 2015.
- [4] Analog.com, "ADI - Analog Dialogue | ADC Architectures", 2015. [Online]. Disponible: <http://www.analog.com/library/analogdialogue/archives/39-06/architecture.html>. [Acedido: 5 - Nov - 2015].
- [5] Synopsys, "Scalable Architectures for Analog IP on Advanced Process Nodes", 2015. [Online]. Disponible: <https://www.synopsys.com/Company/Publications/SynopsysInsight/Pages/Art5-data-converters-IssQ3-13.aspx?cmp=Insight-I3-2013-Art5>. [Acedido: 5 - Nov - 2015].
- [6] D. Johns and K. Martin, *Analog integrated circuit design*. New York: John Wiley & Sons, 1997.
- [7] B. Bechen, D. Weiler, T. Boom and B. Hosticka, "A 10 bit very low-power CMOS SAR-ADC for capacitive micro-mechanical pressure measurement in implants", *Advances in Radio Science*, vol. 4, pp. 243-246, 2006.
- [8] A. Gines Arteaga, E. Peralias and A. Rueda, "Blind Adaptive Estimation of Integral Nonlinear Errors in ADCs Using Arbitrary Input Stimulus", *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 2, pp. 452-461, 2011.
- [9] T. Rabuske, F. Rabuske, J. Fernandes and C. Rodrigues, "An 8-bit 0.35-V 5.04-fJ/Conversion-Step SAR ADC With Background Self-Calibration of Comparator Offset", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 7, pp. 1301-1307, 2015.

ANEXO I – COMPARADOR

As tabelas 2 e 3 apresentam as dimensões dos componentes que constituem o comparador (consultar figura 4.1).

Tabela 2 Dimensões dos mosfets que constituem o comparador

| Mosfet | Comprimento (μm) | Largura (μm) |
|---------------|---|---|
| MP1 | 0.6 | 0.35 |
| MP2 | 0.6 | 0.35 |
| MP3 | 15 | 0.35 |
| MP4 | 15 | 0.35 |
| MP5 | 20 | 0.35 |
| MP6 | 0.4 | 0.35 |
| MN1 | 3 | 0.35 |
| MN2 | 3 | 0.35 |
| MN3 | 3 | 0.35 |
| MN4 | 3 | 0.35 |
| MN5 | 1 | 0.35 |
| MN6 | 1 | 0.35 |
| MN7 | 1 | 0.35 |
| MN8 | 1 | 0.35 |
| MN9 | 0.4 | 0.35 |
| MN10 | 0.4 | 0.35 |
| MN11 | 0.4 | 0.35 |
| MN12 | 0.4 | 0.35 |
| MN13 | 0.4 | 0.35 |
| MN14 | 0.4 | 0.35 |
| MN15 | 0.4 | 0.35 |

Tabela 3 Capacidade e dimensões dos condensadores que constituem o comparador

| Condensador | Capacidade (<i>pF</i>) | Comprimento (μm) | Largura (μm) |
|--------------------|-----------------------------------|---|---|
| Ccal1 | 2.17 | 40 | 62.55 |
| Ccal2 | 2.17 | 40 | 63.55 |

No que diz respeito à área ocupada em *layout* por este bloco (consultar figura 5.1), ela é de aproximadamente 0.0128 mm^2 ($80\mu m \times 160\mu m$), representando assim cerca de 5% do circuito final.

ANEXO II – *PROCESS CORNERS*

A performance de qualquer circuito depois de manufaturado está sempre dependente das condições em que esse mesmo circuito foi fabricado. Apesar dos processos serem cada vez mais rigorosos e exigentes, é praticamente impossível controlar todos os parâmetros associados ao fabrico na perfeição. Pequenas variações no processo de fabrico como concentrações mais ou menos elevadas de certos elementos podem manifestar-se também em variações dos parâmetros do *mosfet* como a sua transcondutância por exemplo [6]. Deste modo, os denominados *process corners* indicam quais os valores máximos e mínimos que todos os parâmetros associados a um qualquer elemento podem adotar.

Dando um foco em particular aos *mosfets*, existem duas variações de *process corners* que são mais vulgarmente utilizadas, sendo elas *slow process corners* e *fast process corners*. Ambas as designações indicam que as condições a que o *mosfet* esteve sujeito na fase de fabrico fizeram com que a sua resposta seja mais lenta (*slow process corners*) ou mais rápida (*fast process corners*) relativamente ao tempo típico.

Em suma, com o auxílio destes valores, e para que seja possível avaliar os seus efeitos na fase de desenvolvimento, são criados modelos que contemplam todas estas variações. Assim, durante a fase de simulação, podem ser avaliados os comportamentos que o mesmo circuito pode adotar nos diferentes *process corners*.

ANEXO III – AMPLIFICADOR

OPERACIONAL

As tabelas 4, 5 e 6 apresentam as dimensões dos componentes que constituem o amplificador operacional (consultar figura 4.5).

Tabela 4 Dimensões dos mosfets que constituem o amplificador operacional

| Mosfet | Comprimento (μm) | Largura (μm) |
|---------------|---|---|
| MP1 | 400 | 0.35 |
| MP2 | 400 | 0.35 |
| MP3 | 1 | 1 |
| MP4 | 1 | 0.35 |
| MP5 | 10 | 1 |
| MP6 | 125 | 1.25 |
| MN1 | 20 | 0.35 |
| MN2 | 20 | 0.35 |
| MN3 | 400 | 0.35 |
| MN4 | 1 | 0.35 |

Tabela 5 Capacidade e dimensão do condensador C_c presente no amplificador operacional

| Condensador | Capacidade (pF) | Comprimento (μm) | Largura (μm) |
|-------------------------|---|---|---|
| C_c | 1.2 | 23 | 60 |

Tabela 6 Resistências e dimensões das resistências que constituem o amplificador operacional

| Resistência | Resistência ($k\Omega$) | Comprimento (μm) | Largura (μm) |
|-------------------------|---|---|---|
| R_c | 0.388 | 13.6 | 2 |
| R | 75 | 112.5 | 2 |

No que diz respeito à área ocupada em *layout* por este bloco (consultar figura 5.2), ela é de aproximadamente 0.01024 mm^2 ($64\mu\text{m} \times 160\mu\text{m}$), representando assim cerca de 4% do circuito final.

Relativamente à topologia em seguidor de tensão utilizada para este *ADC*, para além de duas resistências com valores idênticos foi adicionado um condensador em paralelo com uma das resistências do divisor de tensão. Este elemento extra revelou-se necessário pois quando os comutadores do *DAC* eram rearranjados, eram gerados picos de corrente na saída do amplificador. Devido a este facto, o sistema sofria um desequilíbrio que provocava variações de corrente em ambos os ramos do amplificador que, aliadas aos tamanhos relativamente elevados dos *mosfets* de entrada (MP1 e MP2), originava capacidades também elas inconstantes. Visto que o divisor de tensão protagonizado pelas resistências estava exposto a estas capacidades, a tensão que idealmente seria 1.65 V no ponto intermédio também sofria oscilações. Deste modo, com o auxílio do condensador extra, o tempo de estabilização desta tensão foi reduzido consideravelmente.

A tabela 7 apresenta informações mais detalhadas sobre estes três elementos em particular.

Tabela 7 Dimensões e capacidade/resistência dos elementos adicionais da topologia em seguidor de tensão

| Componente | Capacidade / Resistência | Comprimento (μm) | Largura (μm) |
|-------------------|---------------------------------|---|---|
| R1 e R2 | 10 $\text{k}\Omega$ | 23.5 | 3 |
| C | 350 fF | 39.7 | 10 |

ANEXO IV – BANCO DE CONDENSADORES OU DAC

A tabela 8 apresenta as dimensões dos comutadores que constituem o banco de condensadores (consultar figura 4.11).

Tabela 8 Dimensões dos mosfets que constituem os comutadores presentes no DAC

| Mosfet | Comprimento (μm) | Largura (μm) |
|---------------|---|---|
| MP1 | 12 | 0.35 |
| MP2 | 12 | 0.35 |
| MP3 | 5.5 | 0.35 |
| MN1 | 3 | 0.35 |
| MN2 | 3 | 0.35 |
| MN3 | 3 | 0.35 |
| MN4 | 4.3 | 0.35 |

No que diz respeito à área ocupada em *layout* por este bloco (consultar figura 4.11), ela é de aproximadamente 0.0888 mm^2 ($555\mu m \times 160\mu m$), representando assim cerca de 34% do circuito final.

ANEXO V – MÁQUINA DE ESTADOS

No que diz respeito ao esquemático que a máquina de estados apresenta, ele é constituído por pequenos elementos digitais que a biblioteca da tecnologia de fabrico providencia como portas lógicas e *flip-flops*. Uma vez que todo este bloco foi gerado pelo *software* utilizado, não existiu qualquer tipo de interação em baixo nível. Deste modo, as dimensões de todos os elementos presentes na máquina de estados bem como as suas ligações foram escolhidas sem qualquer interferência direta do *designer*. A única forma de interferir, ainda que de forma indireta, com a conceção do circuito é modificar o ficheiro *Verilog* que espelha o comportamento da máquina de estados e/ou alterar as *timing constraints* associadas ao bloco em questão.

Relativamente ao *layout*, o processo é em tudo semelhante à fase de baixo nível descrita atrás. A principal diferença é a de que nesta fase o *software* utiliza o *layout* e não as representações gráficas de cada um dos elementos digitais que constituem a máquina de estados. Em suma, depois de realizadas todas as ligações e otimizações necessárias, a área ocupada por este bloco é de aproximadamente 0.119 mm^2 ($700\mu\text{m} \times 170\mu\text{m}$), representando assim cerca de 45% do circuito final.