

FILTRO RECURSIVO A CAPACITORES CHAVEADOS DIGITALMENTE
PROGRAMÁVEL POR CONTROLE DE CARGA

Joarez Bastos Monteiro

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Aprovada por:

Prof. Antonio Petraglia, Ph.D.

Prof. Mariane Rembold Petraglia, Ph.D

Prof. Antônio Carlos Moreirão de Queiroz, D.Sc.

Prof. Antônio Carneiro de Mesquita Filho, Dr. D'Etat

Prof. Jacques Szczupak, Ph.D

Prof. Sérgio Bampi, Ph.D

RIO DE JANEIRO, RJ – BRASIL

MARÇO DE 2004

MONTEIRO, JOAREZ BASTOS

Filtro Recursivo a Capacitores Chaveados
Digitalmente Programável por Controle de Carga
[Rio de Janeiro] 2004

XIV, 99p. 29,7 cm (COPPE/UFRJ, D. Sc.,
Engenharia Elétrica, 2004)

Tese - Universidade Federal do Rio de
Janeiro, COPPE

1. Filtro programável
2. Filtro recursivo
3. Filtro a capacitor chaveado
4. Circuito integrado

I. COPPE/UFRJ II. Título (série)

À

minha mulher e filhos, que, dedicando-me incondicional apoio e extrema compreensão, souberam conviver com as minhas indisponibilidades, dedico o êxito deste trabalho.

Agradecimentos

Aos Profs. Antonio Petraglia e Mariane Petraglia, pelo incentivo ao doutoramento e orientação.

Ao Prof. Carlos Azeredo Leme (IST – Instituto Superior Técnico, Lisboa, Portugal) pela sugestão de uso de um DAC para o controle da carga em capacitores.

Ao Prof. Fernando Pinto Baruqui pela atenção com que orientou aspectos práticos importantes na elaboração do projeto.

Ao Prof. Mário Vaz pelo apoio e incentivo no uso da FPGA no dispositivo de teste.

Aos Profs. Carlos José Ribas D'Ávila e Marcelo Werneck por disponibilizarem recursos de seus respectivos laboratórios para a confecção do dispositivo de teste e medições.

À equipe do Laboratório de Microscopia Ótica (DMM/EE), pelas fotos do circuito integrado.

Aos alunos Rodrigo de Carvalho Mendez pelo auxílio na programação da FPGA e Pietro Maris Ferreira pela ajuda importante na confecção do circuito impresso do dispositivo de teste.

Às demais pessoas que direta ou indiretamente contribuíram para o desenvolvimento deste trabalho.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

FILTRO RECURSIVO A CAPACITORES CHAVEADOS DIGITALMENTE
PROGRAMÁVEL POR CONTROLE DE CARGA

Joarez Bastos Monteiro

Março / 2004

Orientadores: Antonio Petraglia

Mariane R. Petraglia

Programa : Engenharia Elétrica

Este trabalho apresenta um novo procedimento de implementação de filtros recursivos do tipo IIR (*infinite impulse response*) a capacitor chaveado, digitalmente programáveis. O emprego da estrutura com número de zeros maior que o número de pólos garante sensibilidade na banda de passagem melhor do que a dos filtros clássicos (Butterworth, Chebyshev, elípticos e Bessel) que atendam às mesmas especificações. A utilização de técnicas de conversão digital-analógica (DAC) permite a programação do filtro pelo controle da carga dos capacitores, eliminando a necessidade de bancos de capacitores que aumentam a área de silício e o custo de fabricação do integrado. Finalmente, a técnica de multiplexação do amplificador operacional, aplicada na realização de módulos FIR (*finite impulse response*) de segunda ordem, permite projetar o filtro com baixo consumo de potência.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

A DIGITALLY PROGRAMABLE SWITCHED-CAPACITOR RECURSIVE FILTER
THROUGH CHARGE CONTROL

Joarez Bastos Monteiro

March / 2004

Advisors: Antonio Petraglia

Mariane R. Petraglia

Department: Electrical Engineering

This paper presents a novel procedure for the implementation of digitally programmable switched-capacitor recursive (IIR) filters. The use of a number of zeros greater than the number of poles assures lower sensitivity in the passband than what is usually achieved with classical filters (Butterworth, Chebyshev, elliptic and Bessel). Digital-to-analog conversion (DAC) techniques allow filter programming by controlling capacitor charge instead of adjusting a capacitor bank, reducing both silicon area and costs. Finally, multiplexing individual operational amplifiers among second order FIR modules reduces the number of required operational amplifiers and, consequently, power dissipation.

Índice

AGRADECIMENTOS	IV
RESUMO	V
ABSTRACT	VI
1 INTRODUÇÃO	1
1.1. Motivação.....	1
1.2. Objetivo.....	2
2 FILTROS A CAPACITORES CHAVEADOS.....	5
2.1. Introdução.....	5
2.2. A Técnica de Capacitor Chaveado.....	6
2.3. Vantagens.....	7
3 ESTRUTURA DO FILTRO	9
3.1. Módulo de Zeros	9
3.2. Módulo de Pólos	11
3.3. Estrutura do Filtro.....	14
4 PROGRAMAÇÃO USANDO DAC	16
4.1. Funcionamento do DAC.....	17
4.2. Estruturas dos Capacitores C_0 e C_4 Usando DAC.....	20
5 PROJETO DO FILTRO	22
5.1. Considerações para Implementação em Circuito Integrado	22
5.2. Estrutura Global do Filtro	25
5.3. Projeto do Amplificador Operacional	30
5.4. Bancos de Capacitores	38
5.5. Outros Componentes	40
5.6. <i>Layout</i> do Filtro	40
5.7. Implementação.....	45

5.8.	Placa de Teste	53
5.9.	Programação	54
6	RESULTADOS EXPERIMENTAIS	55
6.1.	Consumo.....	55
6.2.	Distorção Harmônica.....	56
6.3.	Faixa Dinâmica	57
6.4.	Programabilidade.....	58
6.5.	Resposta em Frequência	61
6.6.	Uniformidade.....	67
7	CONCLUSÕES	69
7.1.	Trabalhos Futuros.....	70
APÊNDICE A	PLACA DE TESTE	71
APÊNDICE B	PROGRAMAÇÃO DE COEFICIENTES	87
REFERÊNCIAS	97

Índice de Figuras

Fig. 1.1: Comparação das respostas em frequência do filtro proposto (8 zeros e 2 pólos) e do filtro elíptico de 5ª ordem.	4
Fig. 2.1: Técnica de capacitor chaveado: resistor simulado.....	6
Fig. 3.1: Mapeamento dos zeros	10
Fig. 3.2: Módulo de zeros.....	10
Fig. 3.3: Módulo de pólos.	12
Fig. 3.4: Mapeamento dos pólos complexos com indicação das regiões de exclusão em função da faixa de α_p escolhida.....	14
Fig. 3.5: Estrutura global do filtro com 2 pólos e 8 zeros	15
Fig. 4.1: Estrutura do DAC e diagrama de tempos.	16
Fig. 4.2: Estrutura de C_4 nos módulos de pólos e zeros.	20
Fig. 4.3: Estrutura de C_0 no módulo de pólos.	21
Fig. 5.1: Estrutura do capacitor C_2 insensível a capacitâncias parasitas.....	23
Fig. 5.2: Estruturas finais dos capacitores de ajuste: (a) C_0 no módulo de pólos; (b) C_4 nos módulos de zeros e pólos.	24
Fig. 5.3: Detalhe de <i>layout</i> mostrando a compensação de capacitâncias parasitas pelo ajuste do comprimento das trilhas.	25
Fig. 5.4: Módulo de zeros: diagrama esquemático em estrutura diferencial.....	26
Fig. 5.5: Módulo de pólos: diagrama esquemático em estrutura diferencial.....	27
Fig. 5.6: Diagrama em blocos do filtro.	28
Fig. 5.7: Módulo seletor: diagrama esquemático.	29
Fig. 5.8: Amplificador operacional de transcondutância (OTA) com estrutura cascode dobrado e regulado.....	31
Fig. 5.9: <i>Common-mode feedback</i> (CMFB) dinâmico: diagrama esquemático.	33
Fig. 5.10: <i>Layout</i> do amp-op.....	37

Fig. 5.11: Módulo de zeros: <i>layout</i> do banco de capacitores com elementos <i>dummies</i> e anel de guarda.....	39
Fig. 5.12: <i>Layout</i> : circuito integrado completo.	42
Fig. 5.13: <i>Layout</i> : módulo de zeros.	43
Fig. 5.14: <i>Layout</i> : módulo de pólos.....	44
Fig. 5.15: Circuito integrado: encapsulamento cerâmico 68 pinos.	45
Fig. 5.16: Foto do banco de capacitores do módulo de zeros.....	48
Fig. 5.17: Foto do amplificador operacional.....	49
Fig. 5.18: Foto do módulo de zeros.	50
Fig. 5.19: Foto do módulo de pólos.	51
Fig. 5.20: Fotografia do circuito integrado: (a) interna; (b) aspecto externo	53
Fig. 6.1: Sinal de saída com 0,99% de THD: normal e com média no tempo.	57
Fig. 6.2: Densidade espectral de ruído (V_{rms} / \sqrt{Hz})	58
Fig. 6.3: Programabilidade do módulo de zeros	59
Fig. 6.4: Programabilidade do módulo de pólos: (a) ajuste de α_p ; (b) ajuste de β	60
Fig. 6.5: Resposta em frequência do filtro 1 com detalhe da banda passante.	62
Fig. 6.6: Resposta em frequência do filtro 2 com detalhe da banda passante.	63
Fig. 6.7: Resposta em frequência do filtro 3 com detalhe da banda passante.	63
Fig. 6.8: Resposta em frequência do filtro 4 com detalhe da banda passante.	64
Fig. 6.9: Filtro passa-altas, gerados pelo espelhamento (transformação $z \rightarrow -z$) das estruturas passa-baixas anteriores: (a) filtro 1; (b) filtro 2; (c) filtro 3 e (d) filtro 4.	66
Fig. 6.10: Uniformidade: (a) comparação da resposta ao degrau de nove CIs; (b) detalhe da resposta transiente.	68
Fig. A.1 : Diagrama em blocos do circuito programado na FPGA.....	72
Fig. A.2: Diagrama esquemático da placa de teste (parte I): fontes de corrente, alimentação e conexões.	73

Fig. A.3: Diagrama esquemático da placa de teste (parte II): interface FPGA/CI.....	74
Fig. A.4: Diagrama esquemático da placa de teste (parte III): chaves DIP para programação do filtro.....	75
Fig. A.5: Diagrama esquemático da placa de teste (parte IV): amp-ops de entrada e saída.	76
Fig. A.6: Foto da placa de teste.....	77

Índice de Tabelas

Tabela 5.1: Amplificador operacional: relação de transistores.....	36
Tabela 5.2: Comparação entre as especificações e os resultados obtidos na simulação com parâmetros da tecnologia AMS CYE 0,8 μ m	38
Tabela 5.3: Identificação dos pinos do circuito integrado.....	46
Tabela 6.1: Consumo de potência.....	56
Tabela 6.2: Distorção harmônica.....	57
Tabela 6.3: Características dos filtros projetados	61
Tabela B.1: Programação do coeficiente β (6 <i>bits</i>)	87
Tabela B.2: Programação do coeficiente α (8 <i>bits</i>).....	88

Glossário

α_z	Coeficiente de ajuste do módulo de zeros
α_p	Coeficiente de ajuste angular do módulo de pólos
β	Coeficiente de ajuste radial do módulo de pólos
b_i	i-ésimo <i>bit</i> de uma palavra binária
CI	Circuito Integrado
CMFB	<i>Common-Mode Feedback</i>
CMOS	<i>Complementary Metal Oxide Semiconductor</i>
DAC	<i>Digital-to-Analog Converter</i>
DSP	<i>Digital Signal Processor</i>
EOC	<i>End of Conversion</i>
f_{ck}	Freqüência do <i>clock</i> (sinal S)
fF	Fento Farad: unidade de capacitância equivalente a 10^{-15} Farad
FIR	<i>Finite Impulse Response</i>
f_N	Freqüência normalizada
FPGA	<i>Field Programmable Gate Array</i>
f_s	Freqüência de amostragem
GB	Produto Ganho-Banda Passante
H(z)	Função de transferência no domínio-z
IIR	<i>Infinite Impulse Response</i>

MF	Margem de Fase
N	Número de <i>bits</i> usados na operação do DAC
n_f	Número de fases
OTA	<i>Operational Transconductance Amplifier</i>
Q	Carga armazenada em capacitor
SOC	<i>Start of Conversion</i>
SR	<i>Slew Rate</i>
T	Período de amostragem
T_{ck}	Período do sinal S
T_ϕ	Duração de uma fase
THD	<i>Total Harmonic Distortion</i>
t_{LIN}	Tempo de resposta linear
T_s	Período de amostragem
t_s	<i>Settling time</i>
t_{SR}	Tempo de duração do <i>slew rate</i>
VDD	Tensão de alimentação positiva
VHDL	<i>VHSIC Hardware Description Language</i>
VHSIC	<i>Very High Speed Integrated Circuits</i>
V_{ref}	Tensão de referência
VSS	Tensão de alimentação negativa

Capítulo 1

Introdução

1.1. Motivação

O constante aumento da densidade de integração na tecnologia CMOS associado às características de baixo consumo de potência, alta velocidade e baixa complexidade dos circuitos a capacitor chaveado [1], [2], têm estimulado a compactação de sistemas de processamento de sinais pela integração de sistemas mistos analógico-digitais. Os circuitos a capacitor chaveado têm sido largamente utilizados em aplicações de filtragem, onde a programabilidade é uma característica que tem atraído a atenção de diversos pesquisadores [3]-[9].

Outra característica desejável para os filtros a capacitor chaveado é a baixa sensibilidade às variações dos coeficientes – razão de capacitores – que depende, entre outros fatores, do tipo de estrutura adotada na implementação do filtro. Em estudo comparativo [10] foi mostrado que estruturas recursivas (IIR) a capacitor chaveado tendo função de transferência com número de pólos menor do que o número de zeros [11], além de maior linearidade na fase [12], apresentam menor sensibilidade às variações de coeficientes do que as realizações *ladder* do filtro elíptico – números de pólos e zeros iguais – que atendam às mesmas especificações.

A programabilidade continua sendo uma das mais importantes características dos circuitos a capacitores chaveados, pelo seu potencial de utilização, seja para tornar o circuito menos dependente das inevitáveis variações do processo de integração, seja para utilização em circuitos programáveis e/ou adaptativos. Os circuitos desenvolvidos

utilizando banco de capacitores para a programação de coeficientes [4], [6], [7], revelam as principais desvantagens desta técnica, tais como a capacitância parasita associada ao grande número de capacitores do banco, a grande área de circuito integrado (CI) destinada aos capacitores e a elevada dispersão destes componentes que contribuem, também, para o aumento do consumo e dificultam a aplicação de técnicas de projeto de CI que garantam a precisão da razão dos capacitores. Estas características, indesejáveis para estruturas em circuito integrado, têm estimulado o aparecimento de técnicas alternativas [3], [5], [13] que utilizam, para programação, o controle de carga em substituição ao banco de capacitores, visando as reduções de consumo e de área de integração como principais vantagens. Numa destas técnicas apresentada em trabalho recente [13], o número de passos de programação desejado é selecionado pela fixação de um índice de programação m , que também estabelece a relação entre as freqüências de *clock* e de amostragem. Por exemplo, para o caso de programação com 8 bits de resolução, a freqüência de *clock* deverá ser 256 vezes maior do que a freqüência de amostragem. No trabalho apresentado nesta tese, utilizando outro tipo de controle de carga [3], [5], este aumento na freqüência de *clock* é de apenas 21 vezes para a mesma resolução e, conseqüentemente, leva a soluções que além das vantagens inerentes à técnica, como menor consumo de potência e de área de silício, possibilita a operação em freqüências mais elevadas.

1.2. Objetivo

O principal objetivo deste trabalho é apresentar uma nova solução [14], [15] para a realização de filtro digitalmente programável a capacitor chaveado, com estrutura de baixo consumo e baixa sensibilidade à variação das razões de capacitores, e, sendo assim, apropriada para a implementação em circuito integrado.

Na programação é empregada uma técnica de conversão digital-analógica (DAC) [3], [5] que permite a seleção dos coeficientes do filtro pelo controle da carga do capacitor, e não pela variação da área como é convencionalmente utilizado. Como conseqüência, além da facilidade de programação, obteve-se uma solução com baixa dispersão de valores de capacitores e com área de integração significativamente reduzida. É importante observar que a baixa dispersão, além de melhorar a precisão das razões de capacitores, exige menor consumo de potência nos amplificadores.

Seções FIR de segunda ordem, obtidas com um único amplificador operacional (amp-op) multiplexado, são usadas na realização dos zeros e pólos do filtro, conduzindo a uma estrutura bastante atraente para a integração por sua modularidade e baixo consumo.

Para mostrar a viabilidade da técnica proposta foram projetados quatro filtros, todos com estrutura fixa de oito zeros sobre a circunferência de raio unitário e apenas dois pólos complexos. A comparação de um destes filtros projetados com um filtro elíptico de quinta ordem revela apenas uma banda de transição ligeiramente maior, conforme se observa na Fig. 1.1. Deve-se observar que as complexidades dos dois filtros são equivalentes uma vez que, em ambos, a soma do número de pólos e zeros é a mesma.

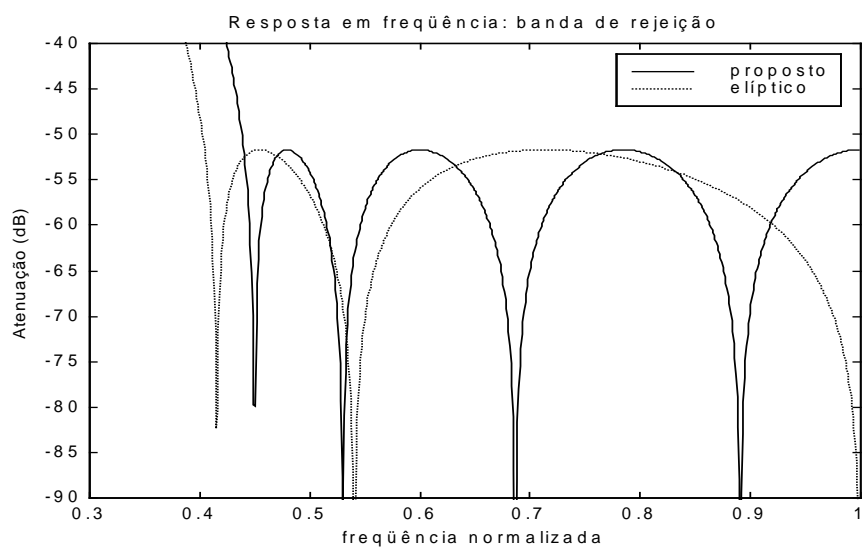
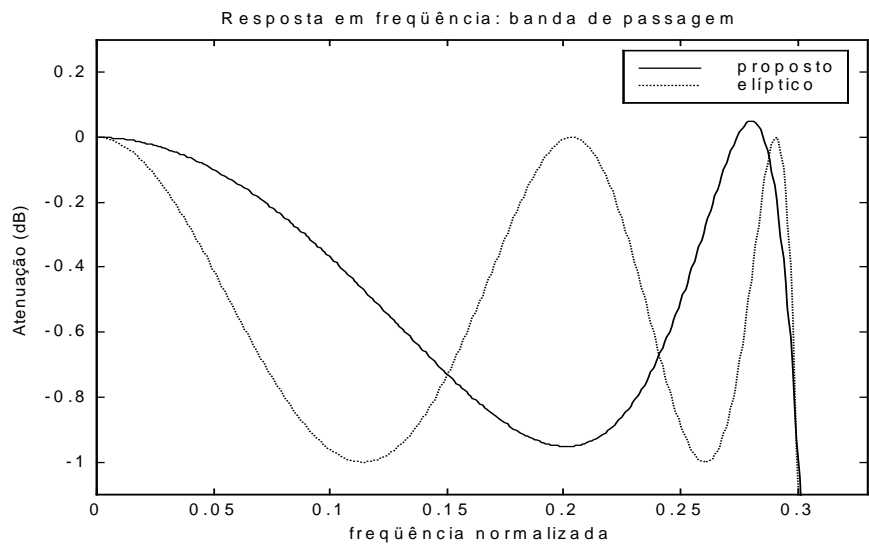
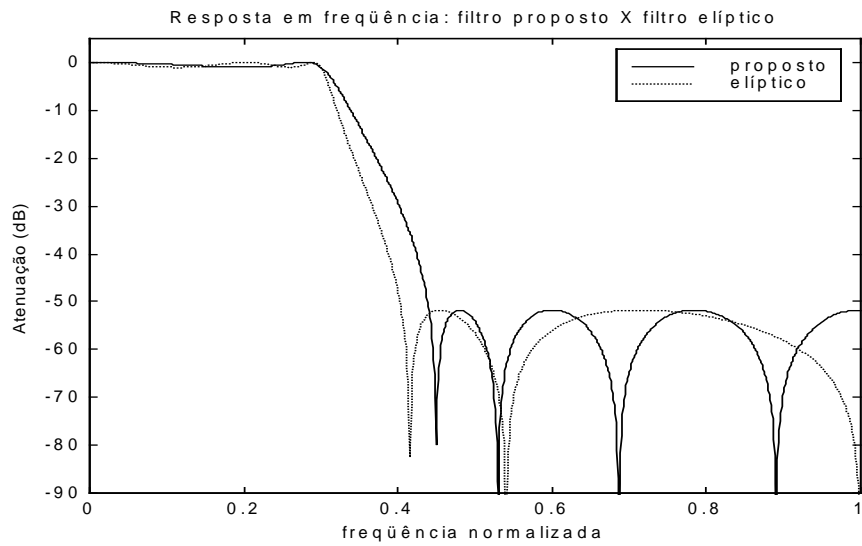


Fig. 1.1: Comparação das respostas em frequência do filtro proposto (8 zeros e 2 pólos) e do filtro elíptico de 5ª ordem.

Capítulo 2

Filtros a Capacitores Chaveados

2.1. Introdução

Os sinais elétricos analógicos são aqueles cujas amplitudes podem assumir quaisquer valores em qualquer intervalo de tempo, ou seja, são sinais contínuos tanto em amplitude como no tempo. Os sinais digitais são representações numéricas (binárias) de sinais analógicos amostrados em intervalos de tempo regulares, isto é, são sinais discretos tanto em amplitude como no tempo. Os sistemas de processamento destes sinais usualmente são classificados como sistemas analógicos ou digitais. Os circuitos RLC, RC-ativos e GmC são exemplos de processadores analógicos de sinais e os computadores de uso geral ou dispositivos dedicados como, por exemplo, um DSP (*Digital Signal Processor*) são exemplos de processadores digitais.

A disseminação da tecnologia MOS de fabricação de circuitos integrados e o surgimento, na década de 70, da técnica de capacitores chaveados [16], impulsionou a utilização de processadores analógicos amostrados, assim denominados por manipularem sinais contínuos em amplitude e discretos no tempo. Os exemplos mais difundidos deste tipo de processadores são os circuitos a capacitores chaveados, cuja importância se deve, principalmente, ao fato de permitirem a integração monolítica de sistemas mistos analógico-digitais, utilizando apenas componentes de fácil integração, como amplificadores, chaves e capacitores.

2.2. A Técnica de Capacitor Chaveado

A idéia principal da técnica de capacitores chaveados [16] consiste na substituição de resistores por capacitores acionados por chaves permitindo, simultaneamente, a redução de área de silício e o aumento da precisão na implementação dos circuitos.

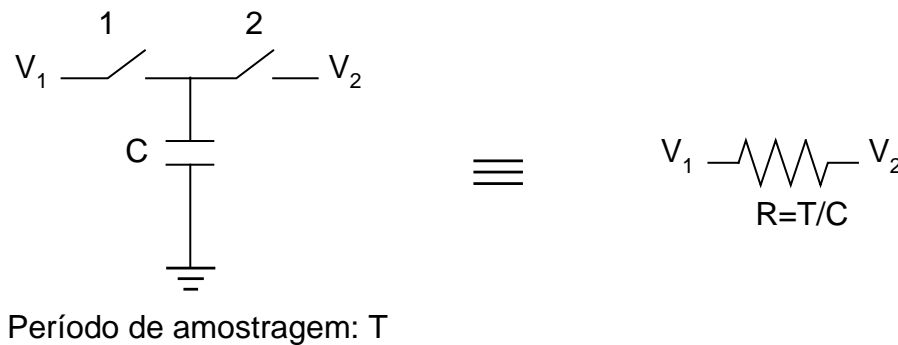


Fig. 2.1: Técnica de capacitor chaveado: resistor simulado

Considere o circuito da Fig. 2.1 onde, na fase 1, a chave 1 é fechada e a chave 2 é aberta e, na fase 2, alternam os respectivos estados. As fases 1 e 2 que comandam o fechamento das chaves são pulsos que se repetem a cada intervalo de tempo T, denominado de período de amostragem. Na fase 1, o capacitor C é carregado com a tensão V_1 . Na transição de fases, ao abrir a chave 1, o capacitor continua armazenando a carga $Q_1 = C \cdot V_1$. Em seguida, ocorre o fechamento da chave 2, e a carga armazenada no capacitor se torna $Q_2 = C \cdot V_2$. Assim, a carga transferida de V_1 até V_2 em cada período de tempo T é dada pela Eq.2.1.

$$\Delta Q = C \cdot (V_1 - V_2) \quad (2.1)$$

Pode-se então definir, pela Eq.2.2, a corrente média transferida de V_1 até V_2 .

$$i \triangleq \frac{\Delta Q}{T} = \frac{C}{T} (V_1 - V_2) \quad (2.2)$$

Por analogia com um circuito resistivo, pode-se dizer que o circuito da Fig. 2.1 se comporta como um resistor $R = T/C$.

2.3. Vantagens

Com a técnica apresentada na Seção 2.2, pode-se concluir que os resistores de um filtro RC-ativo podem ser substituídos por ramos a capacitor chaveado equivalentes. Assim, as constantes de tempo do circuito original serão modificadas com a introdução do resistor simulado. Por exemplo, a constante de tempo R_1C_2 será modificada conforme a Eq.2.3, onde C_1 é o capacitor utilizado para simular R_1 e f_s é a frequência de amostragem, correspondente ao período T .

$$R_1C_2 = (T/C_1)C_2 = (C_1/C_2)f_s \quad (2.3)$$

Como as relações de capacitores podem ser implementadas em tecnologia CMOS com precisão de 0,1%, e a frequência de amostragem pode ser precisamente controlada por um oscilador a cristal, as constantes de tempo no circuito a capacitores chaveados são muito mais precisas do que aquelas do circuito original RC-ativo.

Além da precisão, outra vantagem apresentada pelos circuitos a capacitores chaveados é a grande redução da área necessária para simular um resistor em comparação com a fabricação direta de R . Por exemplo, simulando um resistor de 10 M Ω , usando frequência de amostragem de 100 kHz, a área utilizada no circuito integrado seria cerca de 400 vezes menor [1].

É oportuno lembrar que os circuitos a capacitores chaveados, obtidos por substituição direta dos resistores, exigem que a frequência de amostragem seja muito maior que a maior frequência do sinal processado, para que o filtro opere corretamente. Entretanto, se a função de transferência for produzida diretamente no domínio da transformada-z [1], como é o caso dos circuitos desenvolvidos neste trabalho, não

ocorrerá esta limitação e, desta forma, a maior frequência de operação do filtro ficará limitada a apenas metade da frequência de amostragem (frequência de Nyquist).

A implementação do filtro em módulos de 2ª ordem ligados em cascata permite independência no ajuste dos coeficientes. Desta forma, qualquer erro de realização do coeficiente devido a variações no processo de fabricação fica restrito apenas ao módulo correspondente, evitando a sua propagação por toda a função de transferência. Assim, erros num módulo afetam apenas o posicionamento de um único par de zeros ou pólos complexos conjugados.

Capítulo 3

Estrutura do Filtro

Utilizando a técnica de capacitores chaveados foram desenvolvidos dois circuitos cujas funções de transferência, ambas de 2ª ordem, são capazes de, respectivamente, implementar zeros complexos sobre a circunferência de raio unitário e realizar pólos complexos estáveis.

3.1. Módulo de Zeros

Os zeros da função de transferência são realizados pela associação em cascata de quatro seções FIR de segunda ordem. O módulo de zeros é implementado através da realização de uma função de transferência da forma $1 + \alpha_z z^{-1} + z^{-2}$. O ajuste de um único coeficiente (α_z) no intervalo $[-2, 2]$ permite o posicionamento dos zeros em qualquer ponto da circunferência de raio unitário ($|z|=1$), garantindo o máximo de atenuação na frequência desejada. A Eq. 3.1 permite relacionar o coeficiente α_z com a posição dos zeros em função da frequência normalizada (f_N).

$$\alpha_z = -2 \cos(f_N \pi) \quad (3.1)$$

Na Fig. 3.1 pode ser observado o mapeamento dos zeros correspondendo à programação do coeficiente α_z com palavras de 8 bits.

Contudo, mesmo a implementação deste polinômio de segunda ordem requer estruturas mais elaboradas, de modo a evitar o consumo de potência excessivo devido ao número de amp-ops necessários para a realização da linha de atrasos e dos somadores.

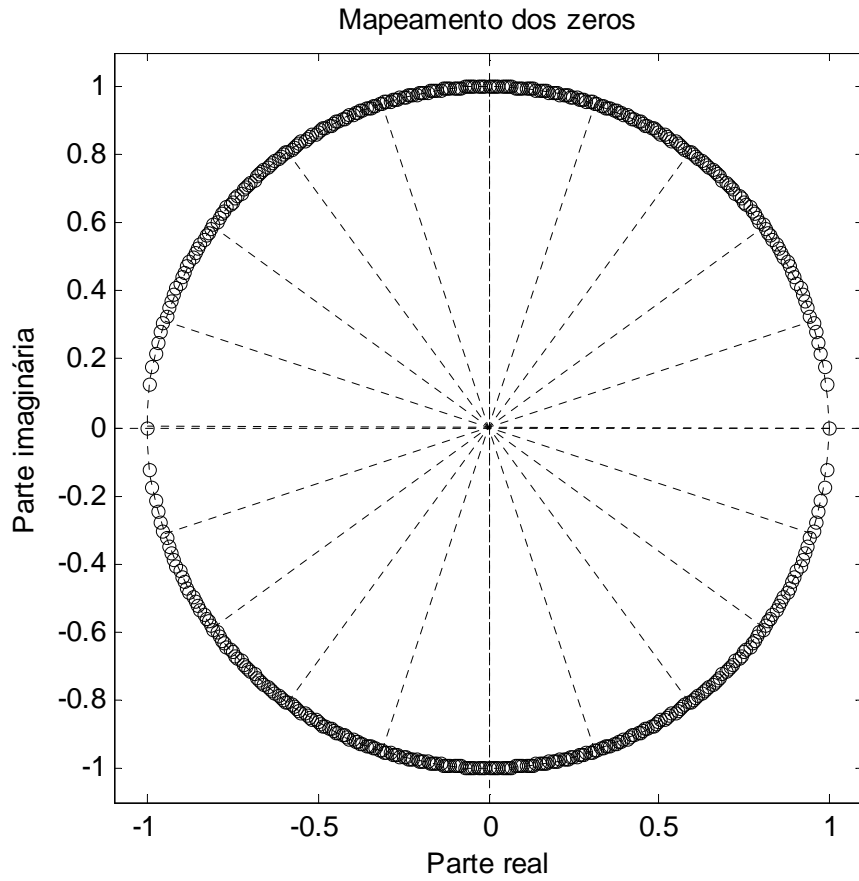


Fig. 3.1: Mapeamento dos zeros

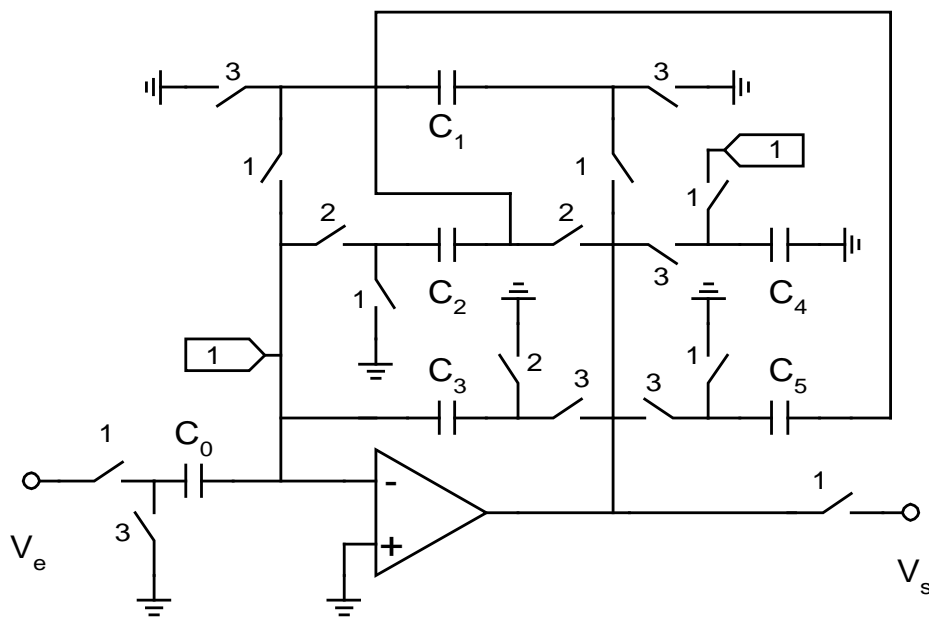


Fig. 3.2: Módulo de zeros.

Fischer [17], [18] apresentou estruturas para implementação de filtros FIR onde um único amp-op é multiplexado de forma a implementar um polinômio de ordem n , com a utilização de $(n+1)$ fases.

Baseado neste estudo foi desenvolvido o circuito da Fig. 3.2 que, utilizando três fases, implementa o polinômio de segunda ordem, cuja função de transferência é apresentada na Eq.3.2, e ainda permite o ajuste do coeficiente de z^{-1} pela variação do capacitor C_4 .

$$H_1(z) = -\frac{C_0}{C_1} \left(1 + \frac{C_4 - C_5}{C_3} z^{-1} + z^{-2} \right) \quad (3.2)$$

Para implementar coeficientes negativos e positivos com a mesma estrutura, foram utilizados dois capacitores (C_4 e C_5) em paralelo na fase de amostragem fazendo-se a inversão de apenas um deles (C_4) na fase de transferência. Assim, o coeficiente $\alpha_z = (C_4 - C_5)/C_3$ é realizado pela diferença entre duas razões de capacitores conforme mostrado em $H_1(z)$, na Eq.3.2. Alterando C_4 para valores acima e abaixo de C_5 , fixado como referência, é possível ajustar coeficientes negativos e positivos sem qualquer modificação da estrutura.

Uma outra característica importante do circuito da Fig. 3.2, que é particularmente desejável para filtros programáveis, é a realização estrutural dos coeficientes unitários dos termos z^0 e z^{-2} , garantindo a localização dos zeros exatamente sobre a circunferência de raio unitário e, conseqüentemente, a máxima atenuação na freqüência desejada, independentemente de qualquer relação de capacitores.

3.2. Módulo de Pólos

O desenvolvimento de um circuito capaz de realizar o par de pólos complexos do filtro, conforme indicado na Fig. 3.3, foi obtido com a associação de um somador e uma

estrutura similar ao módulo de zeros posicionada na malha de realimentação. O polinômio $H_2(z)$, mostrado na Eq.3.3, é obtido com uma pequena modificação no circuito de implementação dos zeros, de modo a eliminar o coeficiente de z^0 .

$$H_2(z) = -\left(\frac{C_4 - C_5}{C_3} \cdot \frac{C_0}{C_1} z^{-1} + \frac{C_0}{C_1} z^{-2}\right) \quad (3.3)$$

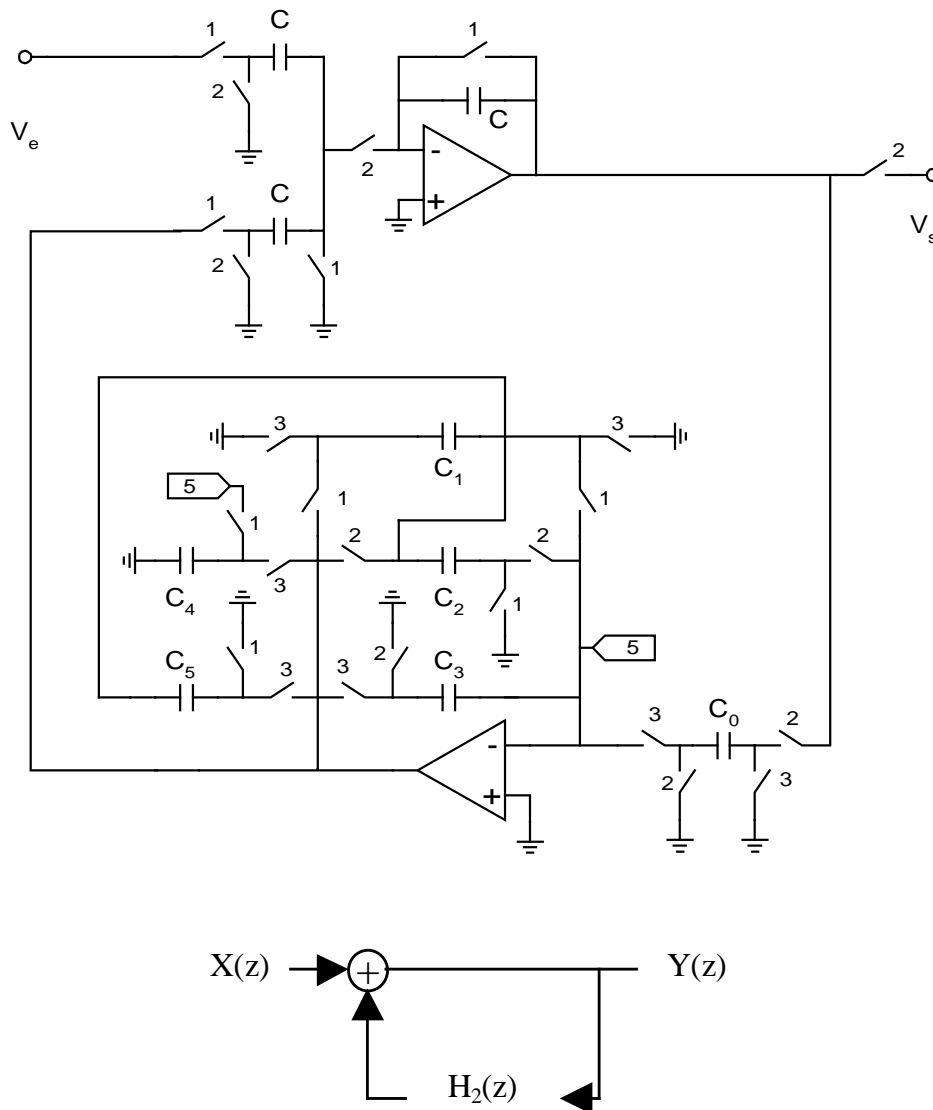


Fig. 3.3: Módulo de pólos.

Com a realimentação é formado o polinômio de segunda ordem no denominador e a função de transferência obtida para o circuito da Fig. 3.3 é apresentada na Eq.3.4.

$$\frac{Y(z)}{X(z)} = \frac{1}{1-H_2(z)} = \frac{1}{1+\alpha_p\beta z^{-1} +\beta z^{-2}} \quad (3.4)$$

As condições para a realização de pólos complexos e para a estabilidade são estabelecidas, respectivamente, pelas Eqs. 3.5 e 3.6.

$$-2\sqrt{\beta} \leq \alpha_p\beta \leq 2\sqrt{\beta} \quad (3.5)$$

$$|z| = \sqrt{\beta} < 1 \Rightarrow 0 \leq \beta < 1 \quad (3.6)$$

O denominador é programado pelo ajuste de C_0 e C_4 que implementam, respectivamente, os coeficientes $\beta = C_0/C_1$ e $\alpha_p = (C_4 - C_5)/C_3$.

Em função da frequência normalizada (f_N), o coeficiente α_p pode ser determinado pela Eq. 3.7 :

$$\alpha_p = -\frac{2 \cos(f_N \pi)}{\sqrt{\beta}} \quad (3.7)$$

A Fig. 3.4 mostra o mapeamento dos pólos complexos, indicando as regiões de exclusão para algumas faixas de variação do coeficiente α_p . Para maior clareza do gráfico, β e α_p foram representados com menor resolução, respectivamente, com 16 e 32 níveis.

É possível projetar o circuito para realizar o coeficiente α_p variando no intervalo $[-m, m]$, adotando $C_4 = m \cdot C_3$ e C_5 no intervalo $[0, 2m \cdot C_3]$. Entretanto, o aumento da carga capacitiva exigiria um amp-op com maior capacidade de corrente de saída. Com o objetivo de utilizar um único projeto de amp-op, foi adotada para α_p a mesma faixa, $[-2, 2]$, de variação de α_z .

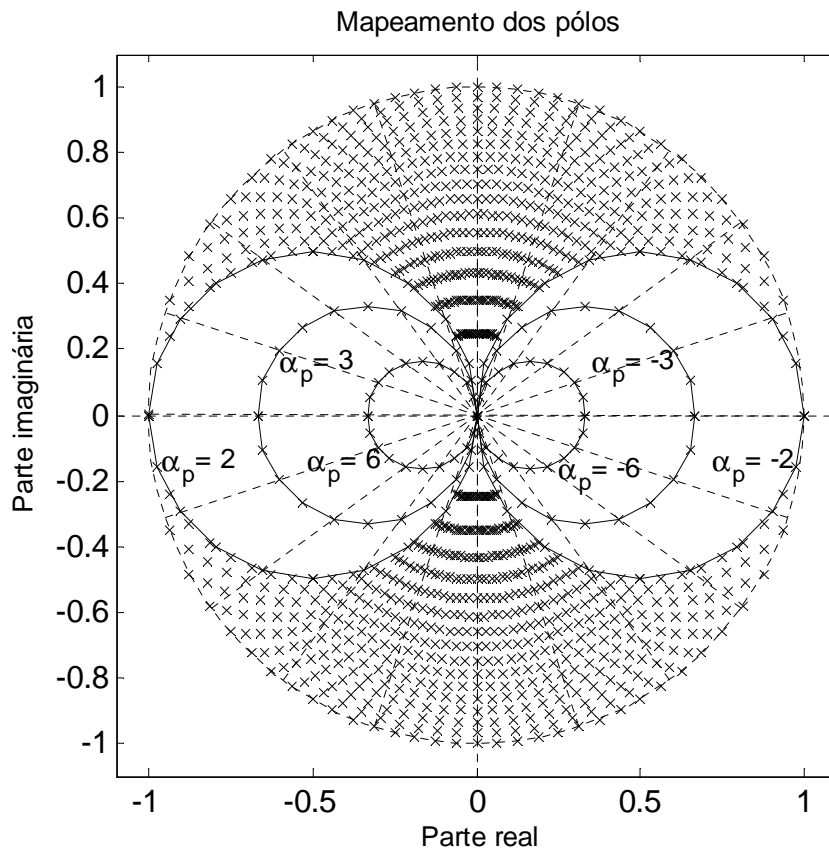


Fig. 3.4: Mapeamento dos pólos complexos com indicação das regiões de exclusão em função da faixa de α_p escolhida.

3.3. Estrutura do Filtro

Na Fig. 3.5 é apresentada a estrutura global do filtro composta de um par de pólos e quatro pares de zeros complexos. Com o ajuste do coeficiente α_z no intervalo $[-2, 2]$, os módulos de zeros podem apresentar ganho DC maior que a unidade (0 dB) para $\alpha_z > -1$. Desta forma, para evitar a saturação do amp-op e, conseqüentemente, a distorção do sinal processado, é previsto um ajuste de ganho para cada um dos quatro módulos de zeros associados em cascata. O módulo de pólos é introduzido no final para que o ganho introduzido pela estrutura compense as perdas geradas pelos módulos de zeros na banda de passagem do filtro. Com esta disposição de módulos e com o ajuste da atenuação adequada de cada módulo de zeros, garante-se que o ganho do filtro seja menor ou igual à unidade, preservando a sua faixa dinâmica.

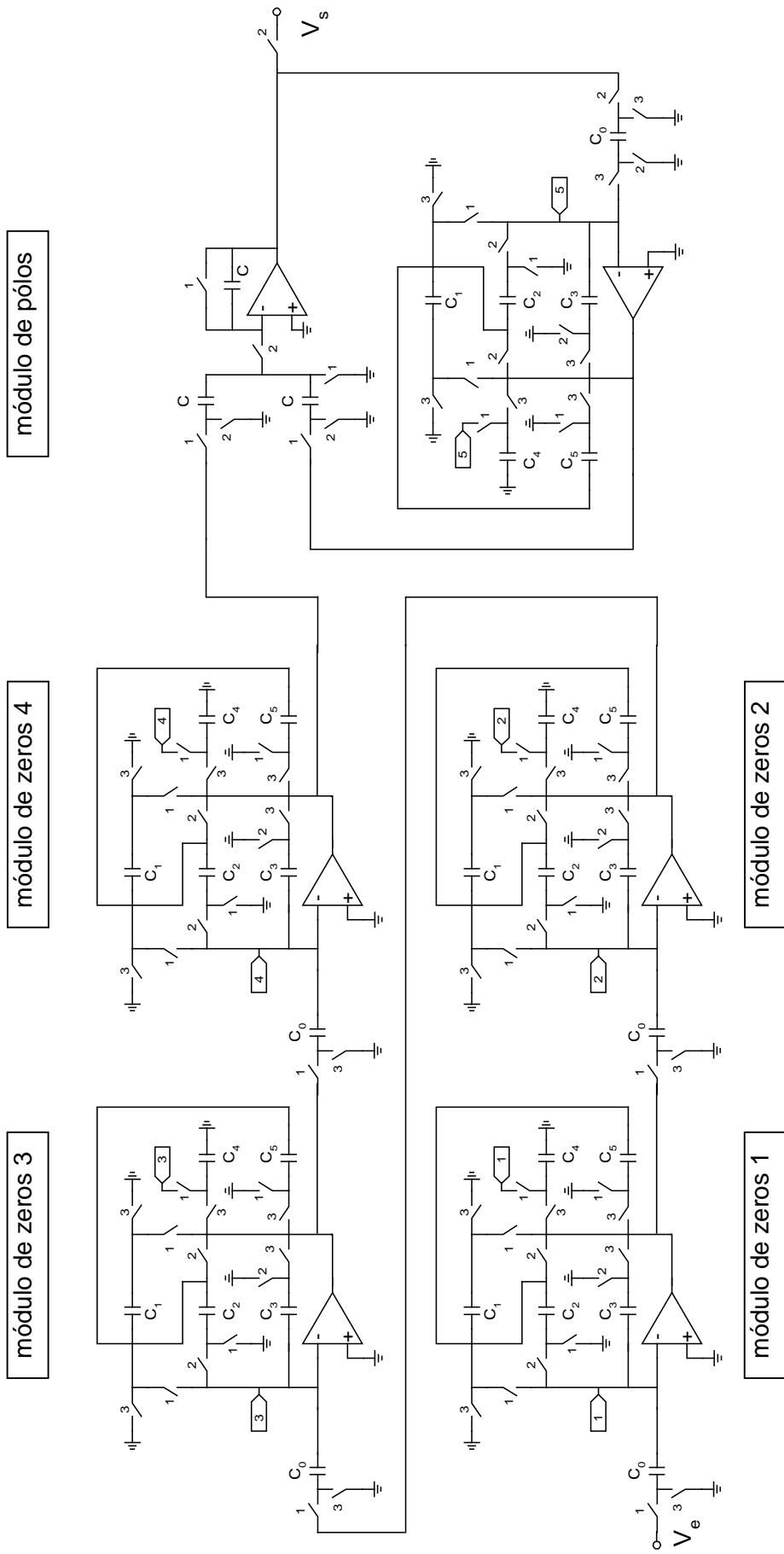


Fig. 3.5: Estrutura global do filtro com 2 pólos e 8 zeros

Capítulo 4

Programação usando DAC

A programação é realizada pelo ajuste dos capacitores C_4 nos módulos de zeros e de C_0 e C_4 no módulo de pólos, de modo a implementar os coeficientes α_z , β e α_p , respectivamente. A forma convencional de programação utiliza bancos de capacitores digitalmente controlados que fazem a variação física dos componentes. As soluções que utilizam esta técnica, além de exigir uma grande dispersão dos valores dos capacitores, ainda ocupam grande área de integração e, conseqüentemente, encarecem o custo de fabricação do integrado.

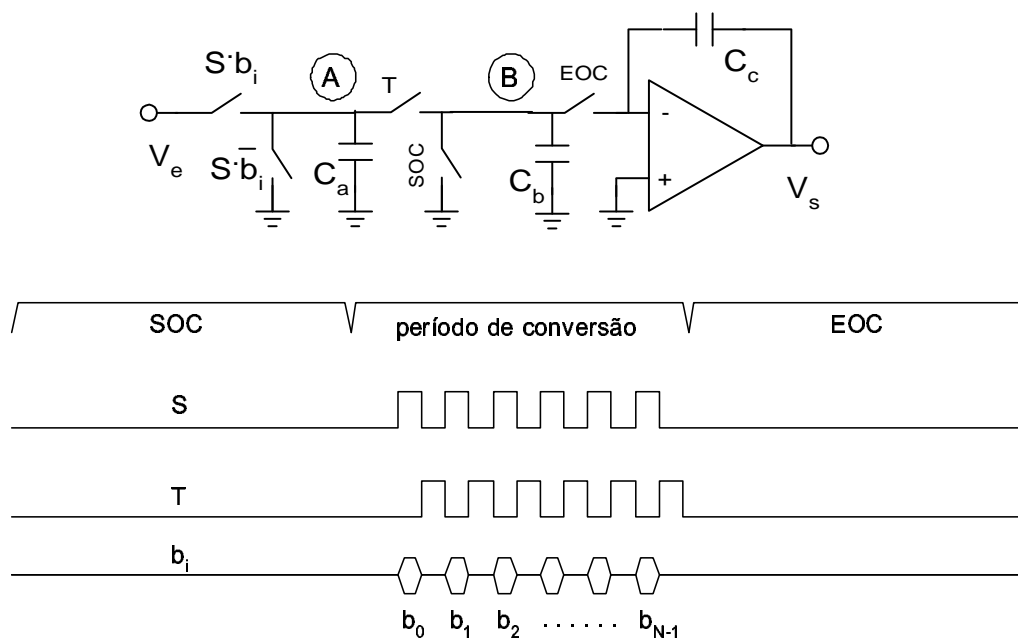


Fig. 4.1: Estrutura do DAC e diagrama de tempos.

Com o objetivo de reduzir a quantidade e a dispersão dos valores dos capacitores, utilizamos uma técnica de conversão digital-analógica (DAC) [3] que simula a variação do valor do capacitor pelo controle de sua carga. A estrutura do DAC e os sinais de controle para o seu funcionamento são apresentados na Fig. 4.1. Pela análise do circuito verifica-se que a tensão final armazenada no capacitor C_b é dependente da tensão de entrada e da palavra binária de N bits aplicada ao sistema.

4.1. Funcionamento do DAC

A duração do processo de conversão é determinada pelo tempo decorrido entre os pulsos de início (SOC – *start of conversion*) e o de final de conversão (EOC – *end of conversion*). Como nesta aplicação o período de conversão corresponde a apenas uma das fases, a fase anterior foi utilizada como pulso de SOC e a posterior como pulso de EOC. Durante o período de conversão, as formas de onda S e T e a palavra binária de N bits comandam o processo de carga do capacitor. Inicialmente, o pulso SOC descarrega o capacitor C_b . Na seqüência, para cada pulso do sinal S as chaves $S \cdot b_i$ e $S \cdot \bar{b}_i$ são acionadas, respectivamente, para $b_i = 1$ e $b_i = 0$, onde b_i é o i -ésimo bit da palavra binária de programação. Se $b_i = 1$, o capacitor C_a é carregado com a tensão de entrada (V_e); se $b_i = 0$, C_a é descarregado. Durante o pulso do sinal T, os capacitores C_a e C_b são conectados em paralelo, redistribuindo as suas cargas. Este processo se repete para cada bit da palavra de programação.

Para cada pulso do sinal S, a carga elétrica acumulada nos capacitores C_a e C_b será:

$$Q_i^{Ca} = CV_e b_i \quad , \quad i = 0,1,2,3\dots(N-1)$$

$$Q_i^{Cb} = \begin{cases} 0 & , \quad i = 0 \\ CV_{i-1} & , \quad i = 1,2,3\dots(N-1) \end{cases}$$

Após cada pulso do sinal T, a carga elétrica armazenada no capacitor equivalente ($C_{eq}=C_a+C_b$) será igual à soma das cargas elétricas de C_a e C_b antes do fechamento da chave T. Assim:

$$Q_i^{C_{eq}} = C_{eq} V_i = Q_i^{C_a} + Q_i^{C_b} \quad (4.1)$$

Pode-se então escrever:

$$\begin{aligned} C_{eq} V_0 &= C V_e b_0 \\ C_{eq} V_1 &= C V_e b_1 + C V_0 \\ &\vdots \\ C_{eq} V_{N-1} &= C V_e b_{N-1} + C V_{N-2} \end{aligned}$$

Considerando $C_a=C_b=C \Rightarrow C_{eq}=2C$, e explicitando V_{N-1} , determina-se a tensão final armazenada no capacitor C_b , que é a mesma do capacitor C_{eq} :

$$V_{N-1} = V_e \frac{\sum_{i=0}^{N-1} 2^i b_i}{2^N} = K V_e \quad (4.2)$$

Ao final da conversão o pulso EOC faz a transferência de carga do capacitor C_b para C_c . Assim, a tensão de saída V_s será:

$$V_s = -\frac{C_b}{C_c} V_{N-1} = -K \frac{C_b}{C_c} V_e \quad (4.3)$$

Portanto, observando a Eq. 4.3, pode-se entender que o efeito final do processo de conversão é a variação do capacitor C_b pelo fator de proporcionalidade K, determinado pela palavra binária pré-programada.

Considerando T_{ck} como o período do sinal S, pelo diagrama de tempos podemos observar que a duração de cada fase corresponde ao número de períodos do sinal S,

equivalente ao número de *bits* (N) utilizados na operação do DAC, mais um período acrescentado no início da fase como tempo de estabilização do sinal na saída do amp-op.

Pode-se então escrever:

$$T_{\phi} = (N + 1)T_{ck}$$

$$T_s = n_f \cdot T_{\phi}$$

e, portanto:

$$T_s = n_f \cdot (N + 1)T_{ck}$$

ou

$$f_{ck} = n_f \cdot (N + 1) \cdot f_s \quad (4.4)$$

onde:

T_{ϕ} - duração de uma fase

T_{ck} - período do sinal S

T_s - período de amostragem

n_f - número de fases

N - número de *bits* usados na operação do DAC

f_{ck} - frequência do *clock* (sinal S)

f_s - frequência de amostragem

Observa-se que a frequência do sinal S (*clock*) aumenta com o número de fases e o número de *bits* utilizados na operação do DAC. Assim, para reduzir a frequência f_{ck} ,

optou-se por usar 6 *bits* na operação do DAC controlando apenas uma parte do capacitor C_4 e, para continuar com uma resolução de 8 *bits*, utilizou-se 2 *bits* adicionais na programação de uma parte fixa deste capacitor, cujos efeitos são somados na fase de transferência de carga para C_1 .

Esta técnica mista de programação dos coeficientes do filtro, parte pela programação de um pequeno banco de capacitores (2 *bits*), parte pelo controle da carga do capacitor (6 *bits*), concilia os dois inconvenientes das duas técnicas que são, respectivamente, o aumento da área de integração e o aumento da frequência de operação. O banco de capacitores de 2 *bits*, formado por capacitores C e $2C$, onde C é o capacitor unitário, não chega a aumentar significativamente a área do circuito integrado e nem mesmo a dispersão dos valores dos capacitores. Contudo, a frequência do *clock* sofre uma redução mais significativa (cerca de 22%) com um *bit* a menos na operação do DAC. Neste caso, a frequência do *clock* (f_{ck}) será 21 vezes maior do que a frequência de amostragem (f_s), conforme se verifica pela Eq. 4.4.

4.2. Estruturas dos Capacitores C_0 e C_4 Usando DAC

Na Fig. 4.2 é apresentada a estrutura que substitui o capacitor C_4 nos módulos de zeros e pólos, de modo a permitir a programação do filtro.

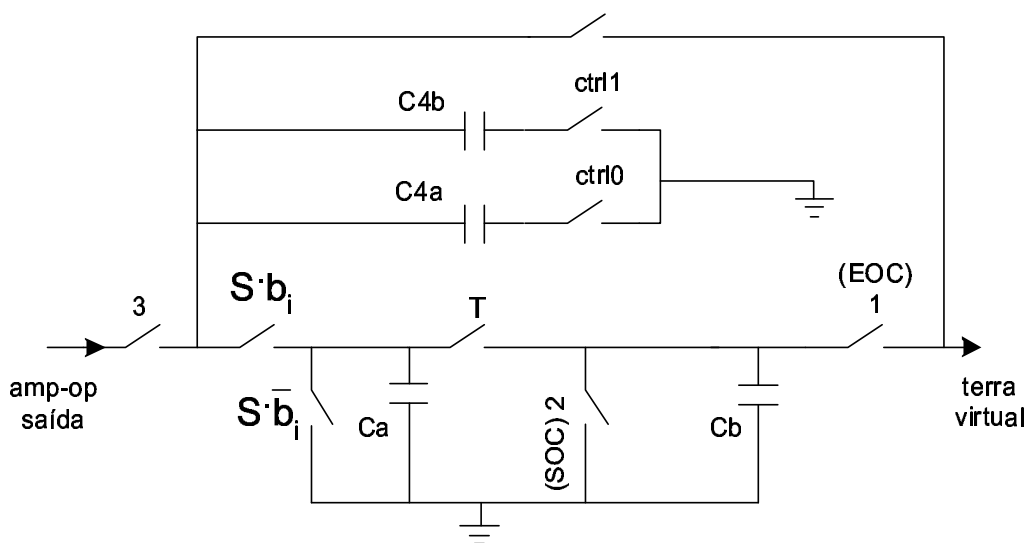


Fig. 4.2: Estrutura de C_4 nos módulos de pólos e zeros.

O pulso de SOC, o período de conversão e o pulso de EOC são, respectivamente, as fases 2, 3 e 1. As chaves *ctrl0* e *ctrl1* implementam os dois *bits* adicionais de controle (banco de capacitores) da parte de C_4 que não é controlada pelo DAC.

Na Fig. 4.3 está a estrutura para o capacitor C_0 do módulo de pólos. O período de conversão ocorre na fase 2, e os pulsos SOC e EOC nas fases 1 e 3. A modificação na posição de C_a é para permitir a inversão de polaridade da tensão de carga do capacitor C_b , necessária para a implementação da função de transferência $H_2(z)$ mostrada na Fig. 3.3. Com o uso do banco de capacitores de 2 *bits*, C_0 poderia assumir um valor máximo de $4C$ e, para manter a função de transferência inalterada, seria necessário redimensionar os demais capacitores (C_1 , C_2 , C_3 , C_4 e C_5). Com a capacitância de carga quatro vezes maior, um dos amp-ops do módulo de pólos deveria ser recalculado para uma corrente de saída maior. Para evitar o aumento de área e de consumo do circuito integrado e o projeto de um novo amp-op, optou-se por suprimir o banco de capacitores de 2 *bits* da estrutura, ficando o posicionamento radial dos pólos (β) ajustável com resolução de 6 *bits*.

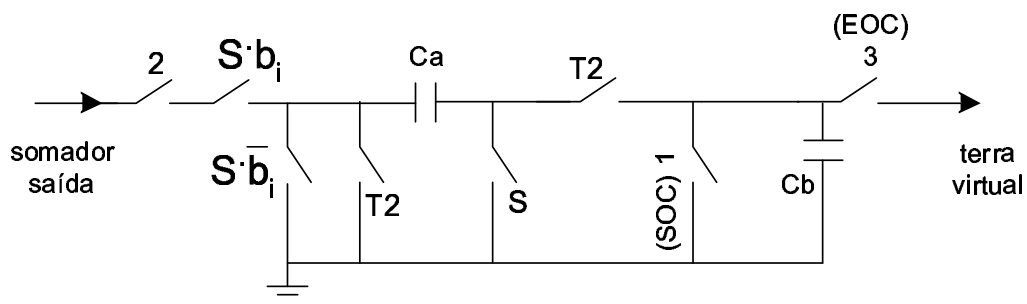


Fig. 4.3: Estrutura de C_0 no módulo de pólos.

Capítulo 5

Projeto do Filtro

5.1. Considerações para Implementação em Circuito Integrado

Para tornar o filtro apropriado para a integração, foram adotadas duas técnicas imprescindíveis para o bom funcionamento do circuito integrado: a estrutura totalmente diferencial e o circuito insensível a capacitâncias parasitas para a terra.

Embora haja um aumento significativo da área de integração em consequência da duplicação do número de chaves e capacitores e do acréscimo no circuito dos amp-ops, a estrutura totalmente diferencial é uma técnica que se torna atraente e muito utilizada pelas vantagens que oferece. Devido à grande atenuação imposta a eventos que ocorrem em modo comum e à duplicação do nível do sinal de saída pelo efeito diferencial, a estrutura utilizada se justifica pelo aumento da faixa dinâmica e redução significativa dos efeitos da injeção de cargas, do “*clock feedthrough*”, bem como da tensão de “*offset*” dos amp-ops [1].

Para que se reduza o custo final e o consumo do circuito integrado, as estruturas desenvolvidas para integração utilizam capacitores, cujos valores, extremamente reduzidos, podem assumir a mesma ordem de grandeza das capacitâncias parasitas das chaves. Desta forma, para que os coeficientes das funções de transferência sejam realizados corretamente, é necessário que se utilize alguma técnica para torná-los insensíveis a capacitâncias espúrias para a terra. As condições estabelecidas por Hasler [19] para tornar o circuito insensível, em geral, introduzem inversão de polaridade que poderiam exigir grandes modificações ou mesmo inviabilizar a solução

numa estrutura unifilar. A topologia totalmente diferencial apresenta mais uma vantagem, pois com a escolha conveniente do terra virtual, permite contornar o problema de sensibilidade a capacitâncias parasitas com pequenas modificações, conforme já observado por outro pesquisador [18].

Para evitar estes problemas de sensibilidade, foram feitas algumas modificações no circuito. Pode-se observar nos módulos de zeros (Fig. 3.2) e de pólos (Fig. 3.3), que o capacitor C_2 está ligado na saída do amp-op na fase 2, comutando para o terra virtual na fase 1. Assim, a carga armazenada na capacitância parasita da chave da fase 2 (da ordem de 50fF) é, também, transferida para o capacitor C_1 . O erro provocado nesta transferência pode chegar a 50%, inviabilizando a realização prática do filtro. Para solucionar este problema as conexões de C_2 foram modificadas conforme mostrado na Fig. 5.1. Deve-se observar que, na fase 1, a carga armazenada na capacitância parasita da chave da fase 2 é descarregada para terra, e somente a carga de C_2 é transferida para C_1 . Para que a tensão em C_1 não fique invertida, o outro terminal de C_2 é ligado ao terra virtual (-), fazendo uso da vantagem da estrutura diferencial.

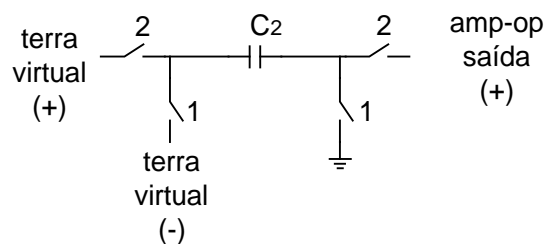
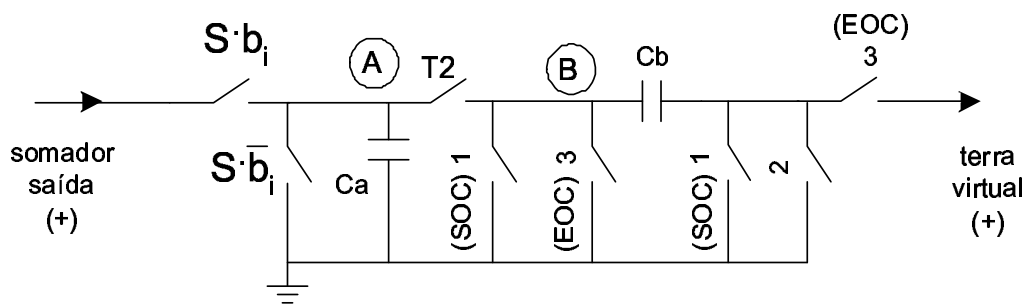


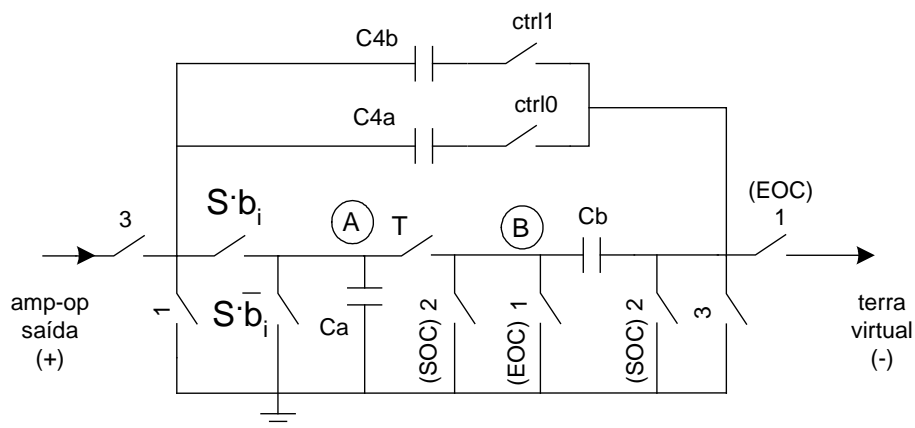
Fig. 5.1: Estrutura do capacitor C_2 insensível a capacitâncias parasitas

Também foi observado que na estrutura do DAC (Fig. 4.1) os nós (A) e (B) são sensíveis a capacitâncias parasitas para a terra, acarretando erros na transferência de carga entre os capacitores C_a , C_b e C_c . Uma vez que a transferência de carga entre C_b e C_c é sensível ao valor absoluto da capacitância parasita, as conexões de C_b foram modificadas de modo a tornar o nó (B) insensível às capacitâncias espúrias das

chaves nele conectadas, na fase de transferência de carga para o capacitor C_c . Na Fig. 5.2 são mostrados os circuitos finais que substituem os capacitores C_0 , no módulo de pólos, e C_4 , de ambos os módulos, para permitir a programabilidade do filtro. Deve-se observar que o circuito para substituição de C_4 apresenta uma inversão de polaridade indesejada, que foi solucionada com a escolha adequada do terra virtual, fazendo uso da característica totalmente diferencial da estrutura.



(a)



(b)

Fig. 5.2: Estruturas finais dos capacitores de ajuste: (a) C_0 no módulo de pólos; (b) C_4 nos módulos de zeros e pólos.

Na transferência de carga entre C_a e C_b a sensibilidade ainda permanece, porém o erro introduzido depende da diferença entre as capacitâncias parasitas associadas a cada capacitor. Neste caso, com uma criteriosa elaboração do *layout*, onde as chaves

associadas aos nós (A) e (B) são posicionadas próximas umas das outras, e o comprimento das trilhas de conexão dos capacitores C_a e C_b às chaves são, rigorosamente, do mesmo comprimento, pode-se garantir o equilíbrio entre as capacitâncias parasitas associadas aos dois nós e, conseqüentemente, o correto funcionamento do circuito. Os detalhes de elaboração do *layout* estão indicados pelas setas na Fig. 5.3.

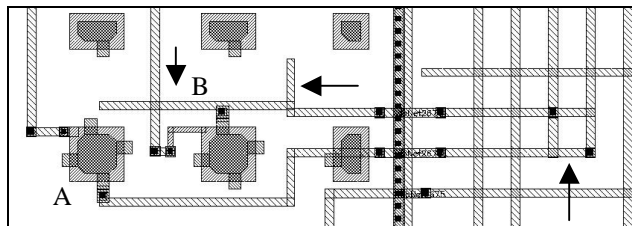


Fig. 5.3: Detalhe de *layout* mostrando a compensação de capacitâncias parasitas pelo ajuste do comprimento das trilhas.

5.2. Estrutura Global do Filtro

Incorporando todas as modificações propostas para os circuitos, nas Figs. 5.4 e 5.5 são apresentados, respectivamente, os esquemáticos dos módulos de zeros e pólos já nas suas formas totalmente diferenciais. Ao lado de cada chave está indicada a fase que provoca o seu fechamento. O conjunto de inversores é responsável pela geração dos sinais complementares, necessários ao funcionamento das chaves. Além dos módulos de zeros e pólos que, associados em cascata, efetivamente formam o filtro, foram adicionados um módulo *buffer* de saída e cinco módulos seletores para permitir a observação dos sinais do interior do circuito integrado, conforme pode ser observado no diagrama em blocos apresentado na Fig. 5.6.

O módulo *buffer* de saída (Fig. 5.6) é formado por capacitores, chaves e quatro amp-ops com ganho unitário, especialmente desenvolvidos [20] para suportar a elevada capacitância (20pF) dos terminais de saída (*pads*) do circuito integrado.

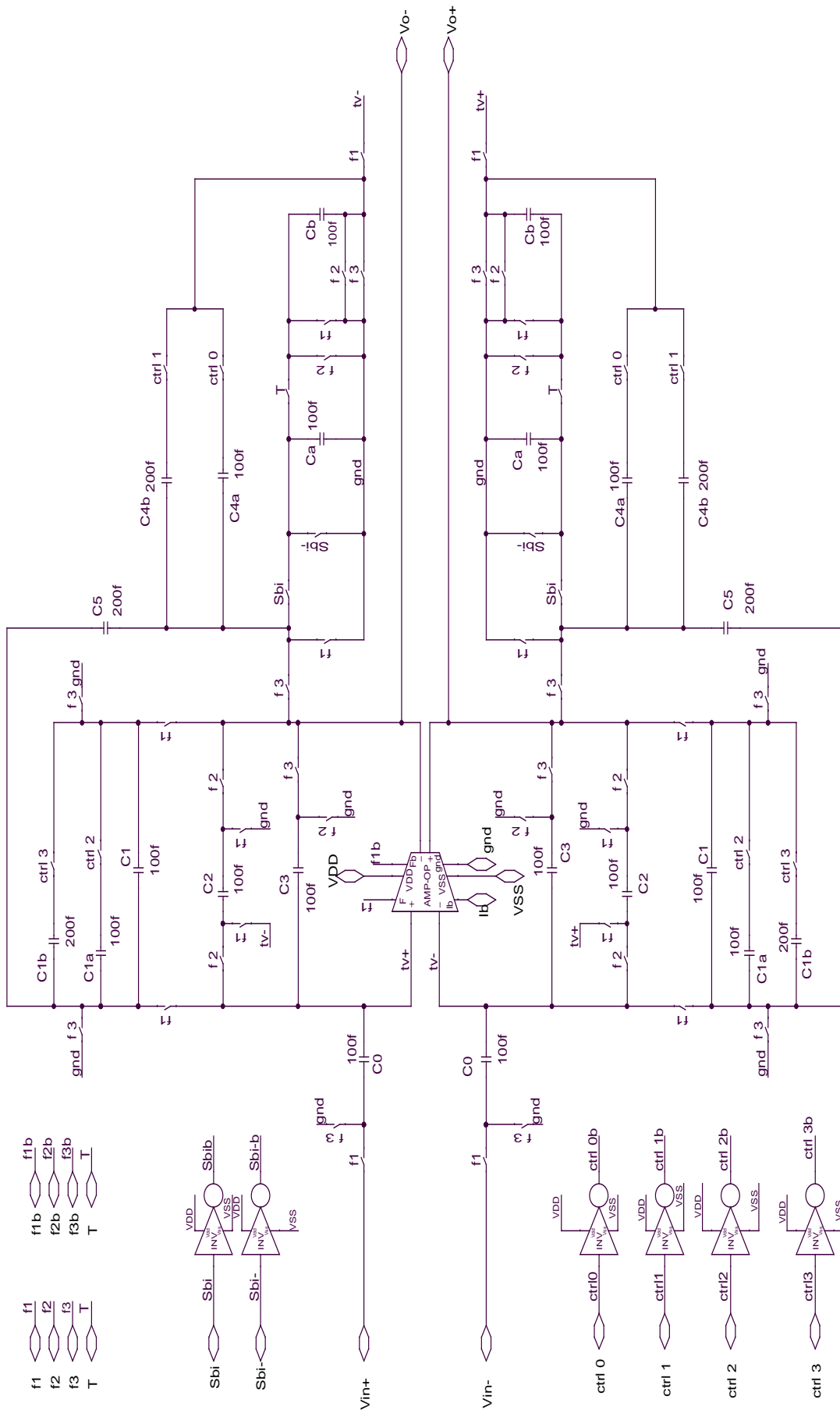


Fig. 5.4: Módulo de zeros: diagrama esquemático em estrutura diferencial.

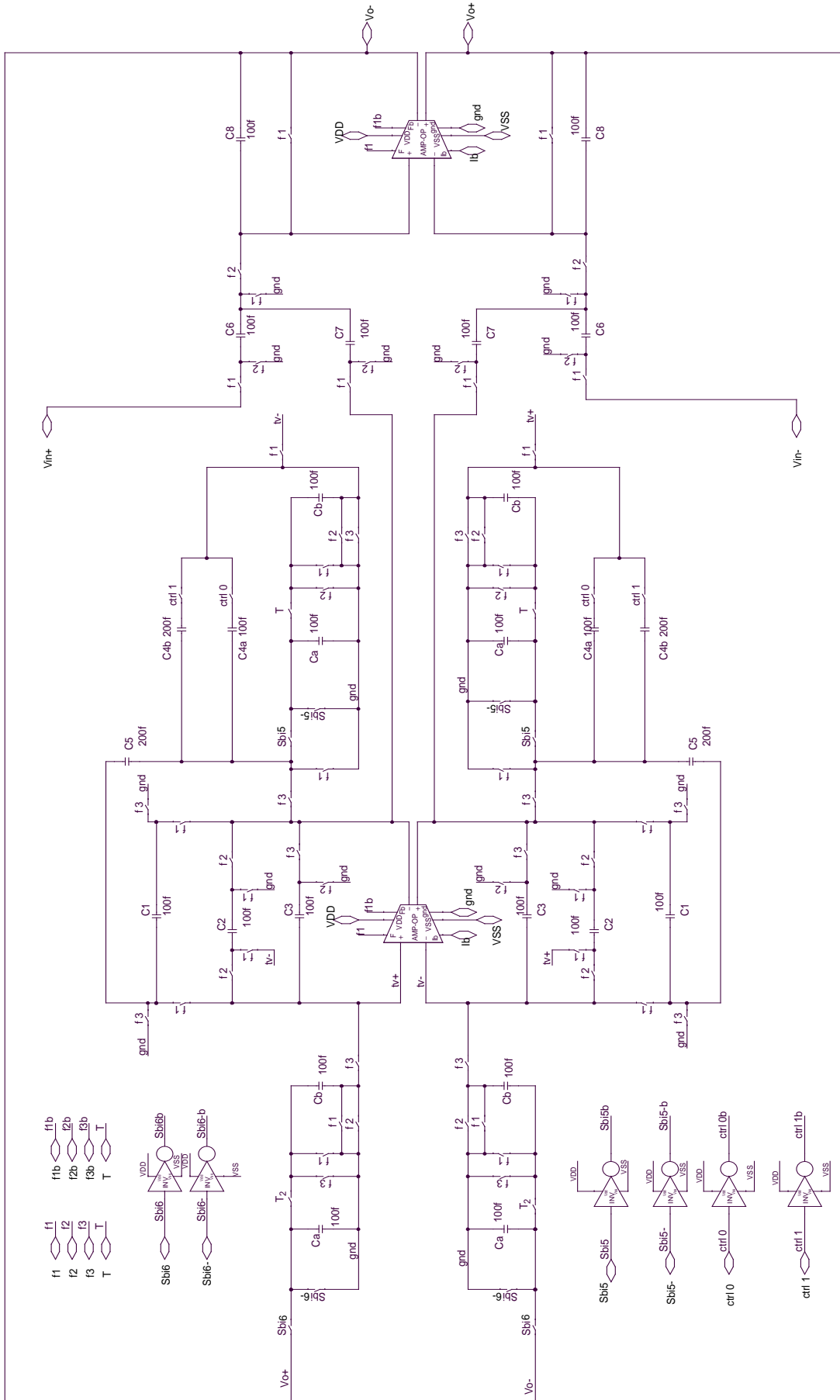


Fig. 5.5: Módulo de pólos: diagrama esquemático em estrutura diferencial.

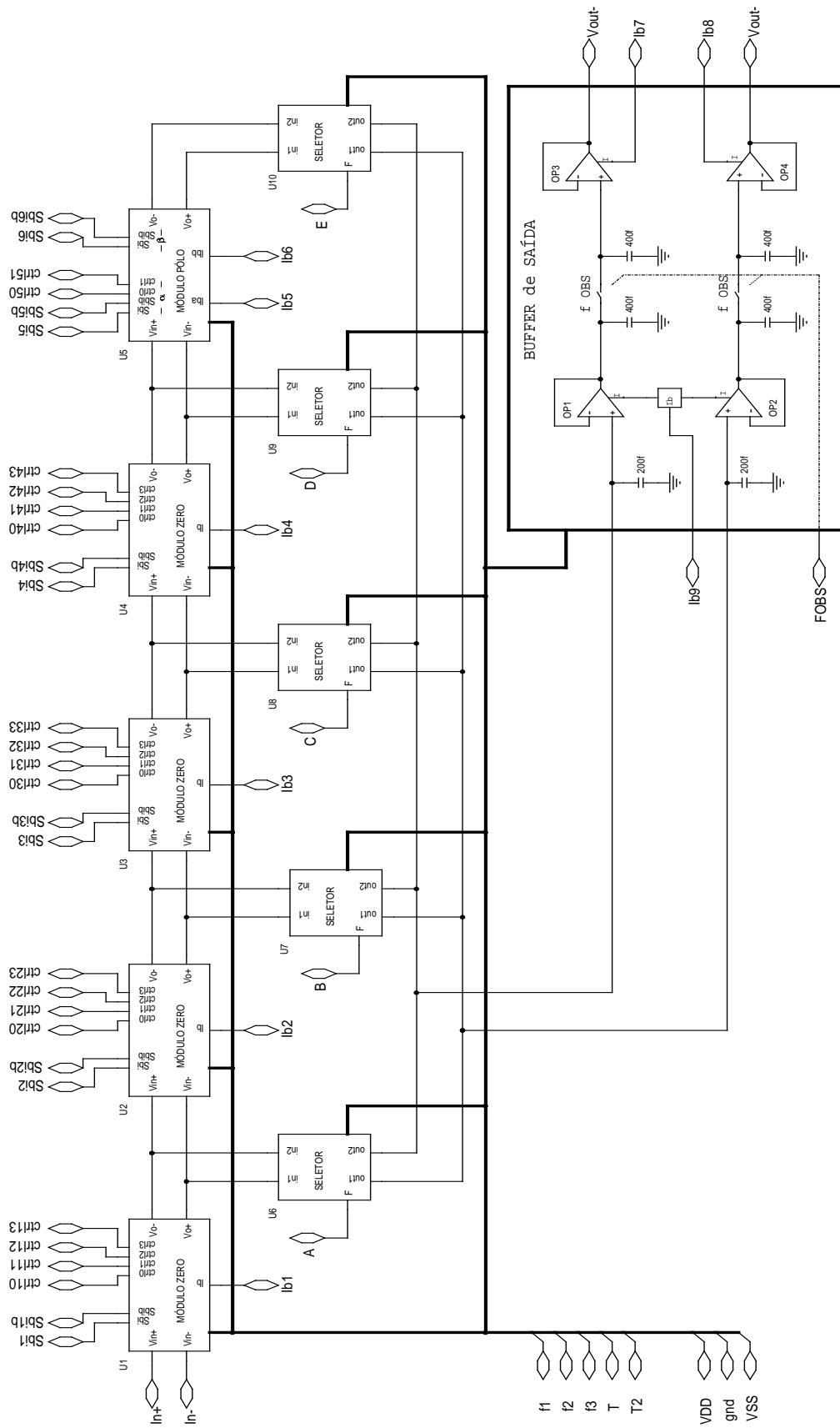


Fig. 5.6: Diagrama em blocos do filtro.

Os módulos seletores são formados pelo conjunto de um inversor e seis chaves, acionadas pelo nível lógico do sinal aplicado na entrada de controle F, conforme mostrado na Fig. 5.7. O nível lógico “1” fecha, simultaneamente, o conjunto de quatro chaves que interligam a saída do módulo selecionado ao *buffer* de saída, permitindo a sua observação. O nível lógico “0”, além de interromper a conexão para a saída, ainda aterriza os nós intermediários. Evita-se, desta forma, que, pelo efeito indesejável das capacitâncias parasitas das chaves, os sinais de saída dos módulos não selecionados sejam acoplados ao sinal observado. Para selecionar uma das saídas, do primeiro ao quarto módulo de zeros, deve-se acionar, respectivamente, uma das entradas A, B, C ou D do circuito integrado, indicadas na Fig. 5.6. O acionamento é feito conectando-se a fase f_1 ao terminal desejado. Para observar a saída do último módulo (resposta do filtro), a fase f_3 deve ser ligada à entrada E do circuito integrado. Para desativar os demais módulos seletores, as respectivas entradas de controle devem ser ligadas ao terminal VSS. A entrada f_{OBS} do módulo buffer opera sempre ligada à fase f_2 .

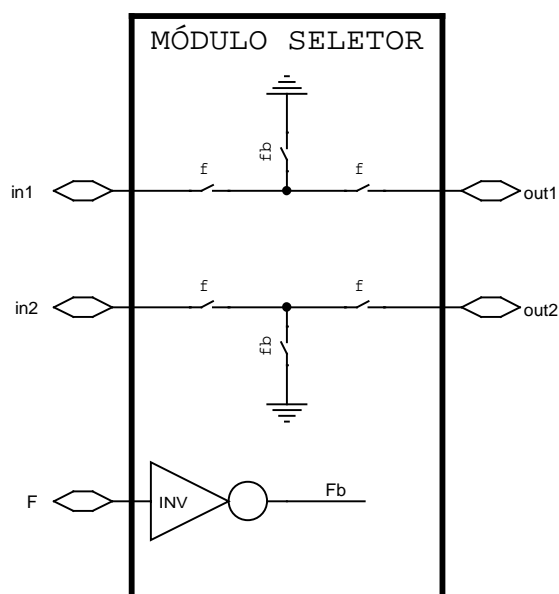


Fig. 5.7: Módulo seletor: diagrama esquemático.

5.3. Projeto do Amplificador Operacional

Para o projeto do amplificador operacional foram adotados os seguintes critérios:

Considerações iniciais

Frequência de amostragem (f_s):	1 MHz
Número de fases (n):	3
Número de <i>bits</i> da programação do DAC (N):	6 <i>bits</i>

Estrutura

Devido ao seu alto ganho, alta resistência de saída, elevado GB e por dispensar compensação interna quando utilizado com cargas capacitivas, optou-se por usar um amplificador operacional de transcondutância (OTA) com estrutura cascode dobrado e regulado [1][20], conforme mostrado na Fig. 5.8.

A estrutura totalmente diferencial necessita que a tensão de modo comum dos terminais de saída seja forçada para o potencial $V_{ref} = (VDD - VSS)/2$, através de um circuito adicional de controle (CMFB – *common-mode feedback*). Neste projeto foi utilizado CMFB dinâmico [21] (Fig. 5.9) para evitar características indesejáveis do circuito de controle contínuo, tais como o elevado consumo de corrente e distorção com sinais de grande amplitude.

CMFB

Pelo circuito da Fig. 5.8 observa-se que as correntes designadas por I_o são determinadas pelos transistores M16 e M28. A tensão de modo comum nas duas

saídas depende do exato equilíbrio entre estas correntes e do perfeito casamento dos transistores. Teoricamente, o projeto do amp-op é feito de tal forma que a tensão $V_b = V_{ctrl}$ garante a corrente I_o e, conseqüentemente, a tensão de modo comum

$$V_{cm} = V_{ref}.$$

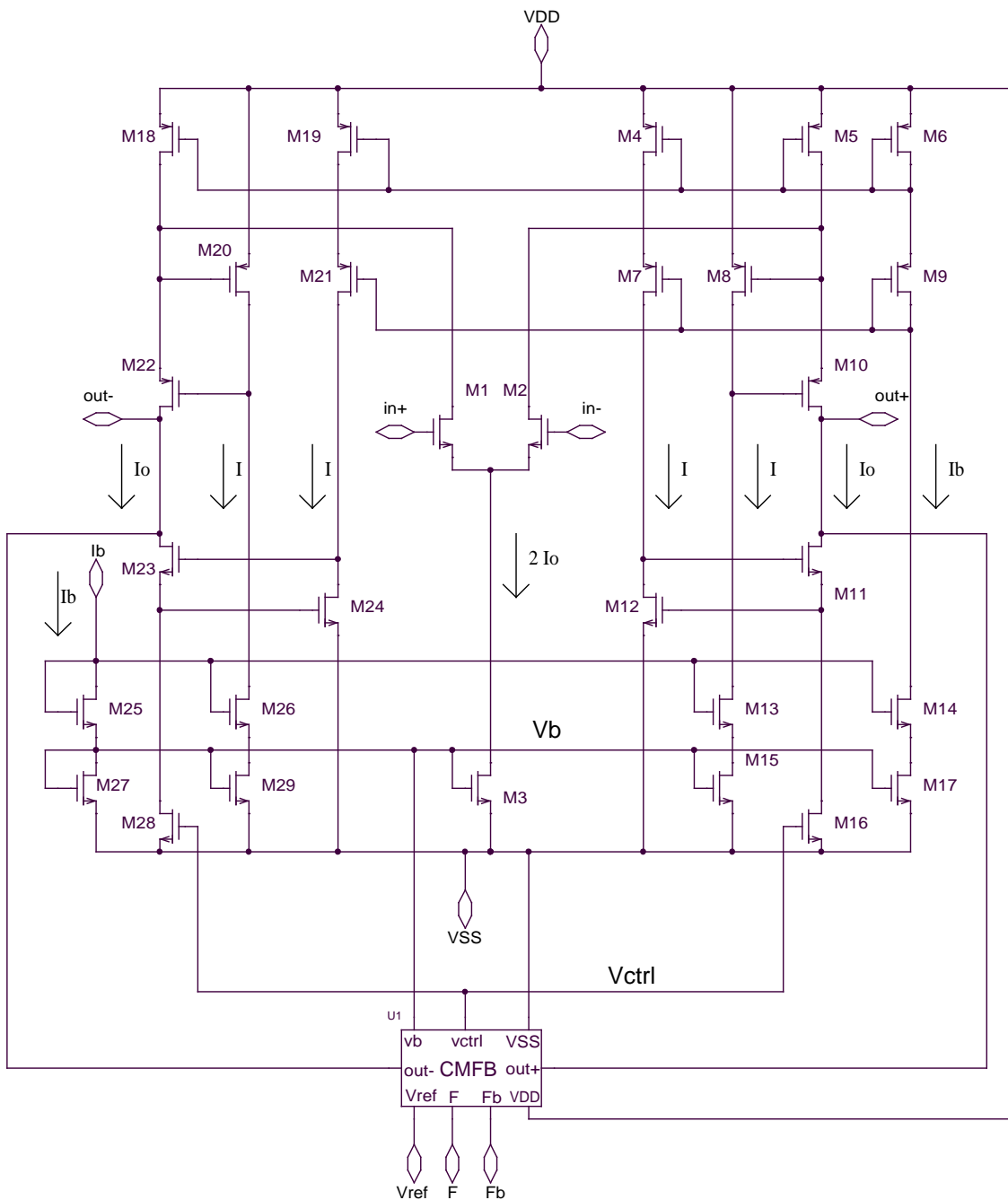


Fig. 5.8: Amplificador operacional de transcondutância (OTA) com estrutura cascode dobrado e regulado.

Na prática, porém, as imperfeições dos transistores e o elevado ganho do circuito, deslocam o valor de V_{cm} , exigindo um circuito de controle para estabilizar a tensão de modo comum, atuando sobre V_{ctrl} .

Para a compreensão do funcionamento do circuito CMFB da Fig. 5.9, deve-se considerar que durante a fase f_{1b} os capacitores C_B são carregados com a tensão $V_{ref} - V_b$. Na fase f_1 , as cargas dos capacitores C_B e C_A são redistribuídas e, na fase f_{1b} seguinte, os capacitores C_A estabelecem a tensão de saída pela soma de V_{ctrl} com a tensão média neles armazenada. Como, na fase f_{1b} , C_B é sempre carregado com a mesma tensão ($V_{ref} - V_b$), fica evidente que após alguns ciclos das fases a tensão média nas saídas tende para o valor de referência V_{ref} , uma vez que $V_{ctrl} \approx V_b$. Deve-se observar nas Figs. 5.8 e 5.9, que qualquer desvio na tensão de modo comum das saídas força a modificação da tensão V_{ctrl} que, por sua vez, altera a polarização dos transistores M16 e M28. Devido ao elevado ganho negativo do circuito as tensões de saída são forçadas a variar em sentido contrário ao desvio inicial. Esta realimentação negativa proporciona um excelente controle da tensão de modo comum da saída. Entretanto, para as excursões do sinal de saída, que constituem variações em modo diferencial, a tensão V_{ctrl} não se altera não havendo, portanto, interferência do circuito de controle de modo comum (CMFB).

Consumo

Pode-se observar na Fig. 5.8 que o consumo de corrente do OTA é dado pela Eq. 5.1.

$$I_{cc} = 4I_o + 2I_b + 4I \quad (5.1)$$

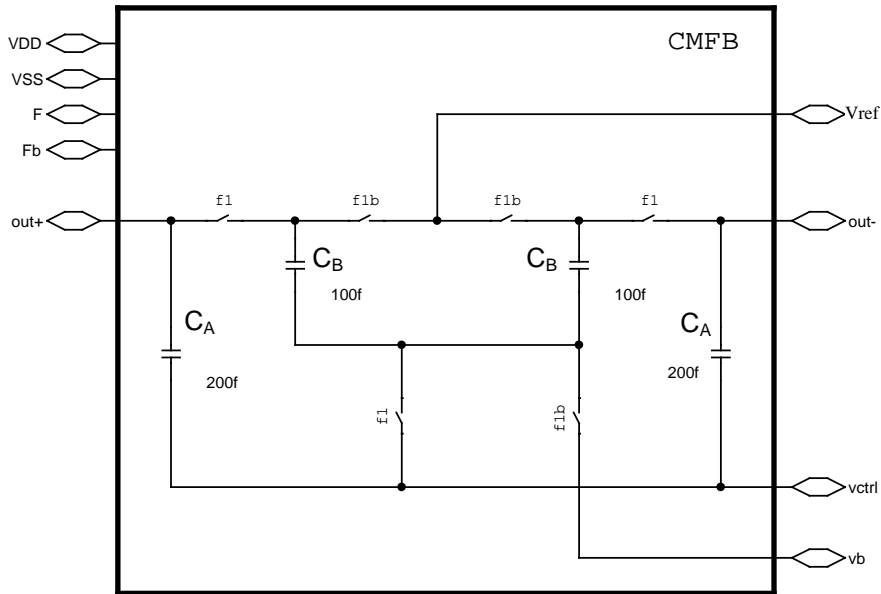


Fig. 5.9: Common-mode feedback (CMFB) dinâmico: diagrama esquemático.

As correntes I e I_b representam acréscimos indesejáveis de consumo e podem ser otimizadas pela escolha adequada das razões de espelhamento $I_o/I_b=M$ e $I_o/I=N$. Assim, o consumo total do amp-op pode ser expresso em função da corrente de saída conforme a Eq. 5.2.

$$I_{cc} = \left(4 + \frac{2(M+2)}{MN} \right) I_o \quad (5.2)$$

Foram adotados $M=N=4$ que, devido à baixa dispersão, permitem um bom casamento entre os transistores dos espelhos de corrente e ainda garante um acréscimo de consumo inferior a 20%. O consumo de corrente de cada amp-op pode ser estimado pela Eq. 5.3.

$$I_{cc} = 4,75I_o \quad (5.3)$$

Resposta transiente

Em cada fase o sinal é amostrado e, para o correto funcionamento do filtro, a tensão de saída deve estar estabilizada antes do final do intervalo de amostragem. O tempo de estabilização corresponde à soma dos tempos de *slew rate* (t_{SR}) e de resposta linear (t_{LIN}) do amplificador. Assumindo que o amp-op se comporta como um sistema de 2ª ordem, a duração da resposta linear (t_{LIN}) pode ser estimada pelo cálculo da diferença entre o tempo de acomodação – *settling time* (t_s) – e o tempo decorrido até a primeira passagem da tensão de saída pelo nível correspondente ao valor final. Manipulando as expressões desenvolvidas para a resposta transiente de sistemas de 2ª ordem [22], e exprimindo em função de GB e MF, obtém-se a Eq. 5.4 para estimativa de t_{LIN} .

$$t_{LIN} = \frac{1}{2\pi GB \cdot 10^{\left(\frac{MF-45}{90}\right)}} \left(\frac{4,6}{\text{sen}(MF/2)} - \frac{\pi \left(90 + \frac{MF}{2}\right)}{180 \cos(MF/2)} \right) \quad (5.4)$$

Foi considerado como tempo de acomodação (t_s), aquele necessário para que o sinal de saída ficasse limitado a um erro menor do que 1% do valor final. Para que a resposta linear do amp-op realimentado tivesse um *overshoot* mínimo e, portanto, estabilizasse mais rapidamente, foi especificado que a margem de fase fosse $MF \geq 60^\circ$.

O menor período de amostragem ao qual o amp-op estará submetido ocorre na operação do DAC e corresponde a 23,81 ns, equivalente ao semi-período da frequência de *clock* (f_{CK}) que é igual a 21 vezes a frequência de amostragem (Eq.4.4). Destinou-se para tempo de estabilização o intervalo correspondente a 75% do tempo de amostragem (cerca de 18 ns) e, visando a redução do consumo, foi adotada a distribuição deste tempo com 10,5 ns para o *slew rate* (t_{SR}) e 7,5 ns para a resposta

linear (t_{LIN}) do amplificador. Para atender aos requisitos de $MF = 60^\circ$ e $t_{LIN} \leq 7,5ns$, o amp-op foi projetado para ter $GB \geq 98MHz$, estimado pela Eq. 5.4.

Quanto ao *slew rate* (SR), considerando como pior situação aquela em que o amp-op deve ser capaz de excursionar $2V_{op}$ com folga de 10% (2,2V) durante o tempo t_{SR} (10,5ns), pode-se especificar o valor $SR \geq 209,5 V/\mu s$.

Corrente de saída

A corrente de saída do amp-op é determinada em função da carga capacitiva (C_L) e do tempo necessário para a estabilização do sinal na fase de amostragem. Resultados satisfatórios na integração de filtro a capacitor chaveado utilizando o capacitor mínimo realizável na tecnologia adotada AMS CYE 0.8 μm [20], estimulou a utilização neste trabalho de capacitor unitário de 100fF. Pelos circuitos dos módulos de zeros e pólos, verifica-se que a capacitância máxima de carga (C_{LMAX}) do amp-op é de 900fF na fase 3, já considerando os capacitores do circuito CMFB. Assim, pela Eq. 5.5, podemos estimar a corrente de saída como $I_o \geq 188,6\mu A$.

$$I_o = SR \cdot C_{LMAX} \quad (5.5)$$

Ganho

Simulações indicaram que para tornar desprezíveis os efeitos do ganho DC finito, o amplificador operacional deve ser projetado com ganho superior a 5000 (74 dB).

Especificações

Resumindo, o amp-op foi projetado para atender às seguintes especificações:

$$\text{ganho DC} \geq 74 \text{ dB}$$

$$\text{GB} \geq 98 \text{ MHz}$$

$$\text{MF} \geq 60^\circ$$

$$I_o \geq 188,6 \mu\text{A}$$

As dimensões dos transistores e o *layout* do amp-op são apresentados, respectivamente, na Tabela 5.1 e Fig. 5.10.

Tabela 5.1: Amplificador operacional: relação de transistores

Transistores	W(μm)	L(μm)
M1, M2	10	1,0
M4, M7, M10, M11, M19, M21, M22, M23	10	0,8
M5, M18	320	0,8
M6, M9	40	0,8
M8, M12, M13, M15, M20, M24, M26, M29	10	1,6
M14, M17, M25, M27	40	1,6
M16, M28	160	1,6
M3	320	1,6

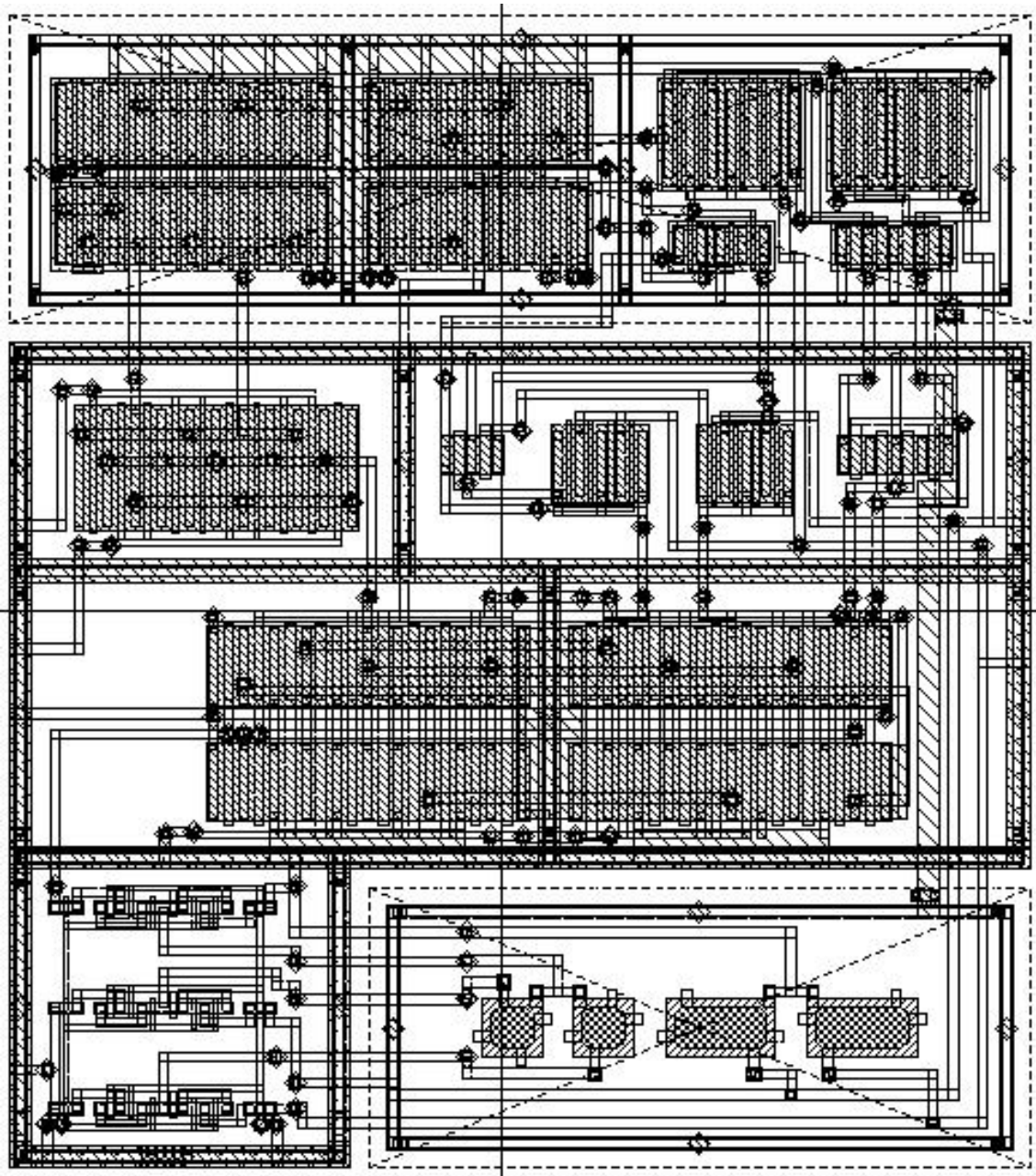


Fig. 5.10: Layout do amp-op.

Simulações

As simulações com o amp-op projetado, usando parâmetros do processo de fabricação da AMS para tecnologia CMOS de $0.8\mu\text{m}$ (AMS CYE $0.8\mu\text{m}$), revelaram características superiores ao especificado, conforme mostra o quadro comparativo da Tabela 5.2. Estes resultados se mostraram promissores, viabilizando a utilização da técnica proposta na implementação de circuitos integrados.

Tabela 5.2: Comparação entre as especificações e os resultados obtidos na simulação com parâmetros da tecnologia AMS CYE 0,8 μ m

	Especificado	Simulado
ganho DC	≥ 74 dB	107 dB
GB	≥ 98 MHz	110 MHz
MF	$\geq 60^\circ$	77°
Io	$\geq 188,6\mu$ A	200μ A

5.4. Bancos de Capacitores

O funcionamento de circuitos a capacitores chaveados é baseado na relação de capacitores que vão implementar os coeficientes da função de transferência desejada. Para garantir o melhor casamento entre os capacitores, estes foram agrupados de forma a aproximar fisicamente aqueles cuja relação de capacitância fosse funcionalmente relevante no circuito. Numa primeira análise, a estrutura diferencial sugere a utilização de dois bancos de capacitores dispostos simetricamente em relação ao amp-op de cada módulo (zero ou pólo). Entretanto, devido à utilização cruzada dos terras virtuais para solucionar os problemas abordados na Seção 5.1, alguns capacitores de um lado têm relação de transferência com outros do lado oposto e, para garantir a precisão dos coeficientes, todos os capacitores foram agrupados num único banco por módulo.

A realização de cada capacitor foi feita pelo agrupamento de células unitárias, cujo valor nominal de 100fF é o mínimo permitido dentro das regras de projeto da tecnologia adotada. O capacitor unitário é construído pela superposição das placas de POLY_1 e POLY_2.

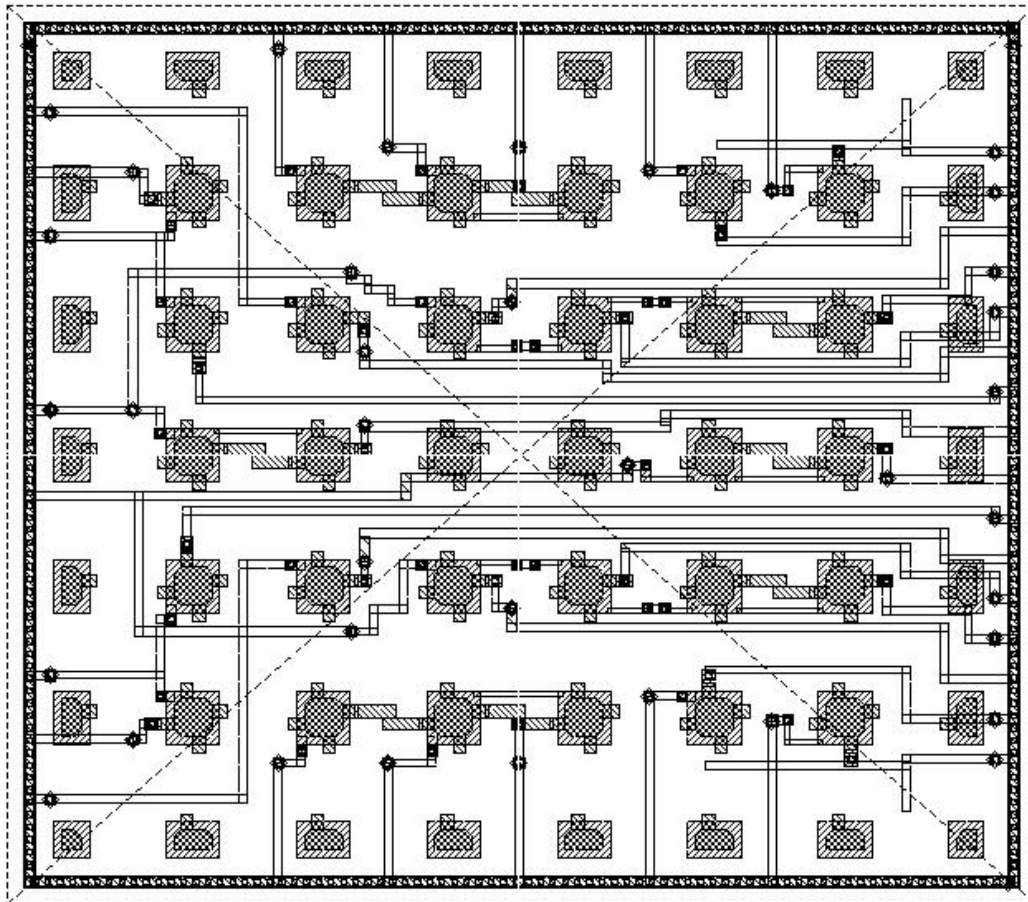


Fig. 5.11: Módulo de zeros: *layout* do banco de capacitores com elementos *dummies* e anel de guarda.

Desta forma, o seu valor é proporcional à área comum das duas camadas. A baixa sensibilidade da capacitância em relação a desalinhamentos das máscaras (deslocamentos e/ou rotações) do processo de fabricação é conseguida com a introdução de pequenas “orelhas” na placa de POLY_2, que mantém constante a área comum [20]. Outras técnicas como anel de guarda polarizado, para evitar acoplamento de sinais espúrios pelo substrato, e elementos *dummies* para equilibrar a estrutura e melhorar a precisão, também foram utilizadas na realização dos bancos de capacitores. Na Fig. 5.11 é mostrado o *layout* do banco de capacitores utilizado nos módulos de zeros.

5.5. Outros Componentes

Para a realização do filtro, além dos amp-ops e capacitores são necessários outros componentes como chaves analógicas responsáveis pela carga e descarga dos capacitores; inversores para a geração de fases invertidas necessárias para o acionamento das chaves; e amp-ops especiais, com grande capacidade de corrente na saída, capazes de suportar as elevadas capacitâncias de carga (20pF) dos terminais de saída do circuito integrado (*pads*). Desta forma, estes amp-ops são utilizados como interface entre a saída interna do filtro e o meio exterior, permitindo a observação do sinal. Neste projeto, as chaves, os inversores e os amp-ops de saída foram os mesmos projetados e utilizados com êxito em trabalho anterior [20], e que fazem parte da biblioteca de componentes do Laboratório de Processamento Analógico e Digital de Sinais (PADS), do Programa de Engenharia Elétrica da COPPE/UFRJ.

5.6. *Layout* do Filtro

O *layout* do filtro consiste na conexão dos diversos componentes – amp-ops, chaves, capacitores e inversores – de modo a realizar o circuito, cujo diagrama em blocos é mostrado na Fig. 5.6. Embora a ligação entre estes elementos seja uma operação simples, cuidados especiais no posicionamento de componentes e trilhas são procedimentos de vital importância para o êxito no funcionamento do circuito integrado.

Devido às dimensões mínimas dos afastamentos entre elementos e trilhas, deve-se adotar alguns cuidados para evitar que o acoplamento entre as diversas partes do circuito, facilitado pelas capacitâncias parasitas, deteriore a qualidade do sinal processado.

Devido à sua estrutura modular, o *layout* de cada módulo foi desenvolvido separadamente e cercado por anel de guarda polarizado, para evitar a interferência de sinais espúrios pelo substrato. Os elementos de cada módulo foram arrumados em camadas, de modo a separar os sinais digitais de controle (fases) do sinal analógico processado. O mesmo cuidado foi tomado nas ligações das chaves aos capacitores e amp-op, evitando o seu cruzamento com as linhas digitais. As entradas e saídas, incluindo as barras de alimentação e terra de cada módulo, foram alinhadas para facilitar a interligação.

Optou-se, ainda, por realizar o circuito integrado com entrada e saída diferenciais, para que se pudesse observar e investigar, separadamente, cada lado do circuito, caso algum problema ocorresse. A conversão, de unifilar para diferencial na entrada e de diferencial para unifilar na saída, foi feita com circuito adicional na placa de teste do circuito integrado.

As fases de controle foram geradas externamente, para evitar que algum problema na parte digital do circuito comprometesse o funcionamento do filtro, e inviabilizasse a comprovação do funcionamento das técnicas investigadas neste trabalho.

O filtro completo, cujo *layout* é apresentado na Fig. 5.12, assumiu as dimensões de 3,0mm por 3,6mm (10,8mm²), incluindo os terminais de saída (*pads*). Os *layouts* dos módulos de zeros e de pólos estão ampliados, respectivamente, nas Figs. 5.13 e 5.14.

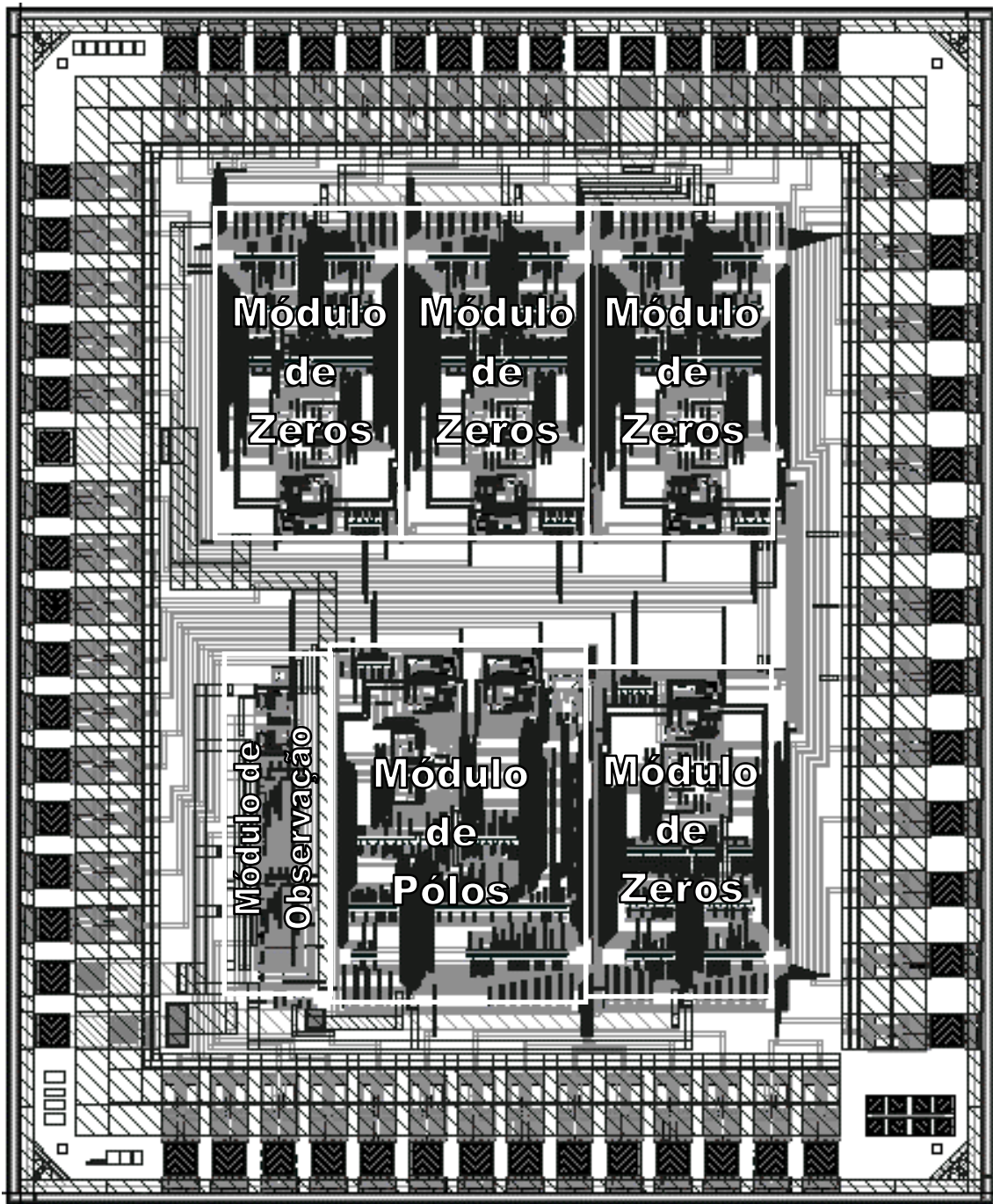


Fig. 5.12: *Layout*: circuito integrado completo.

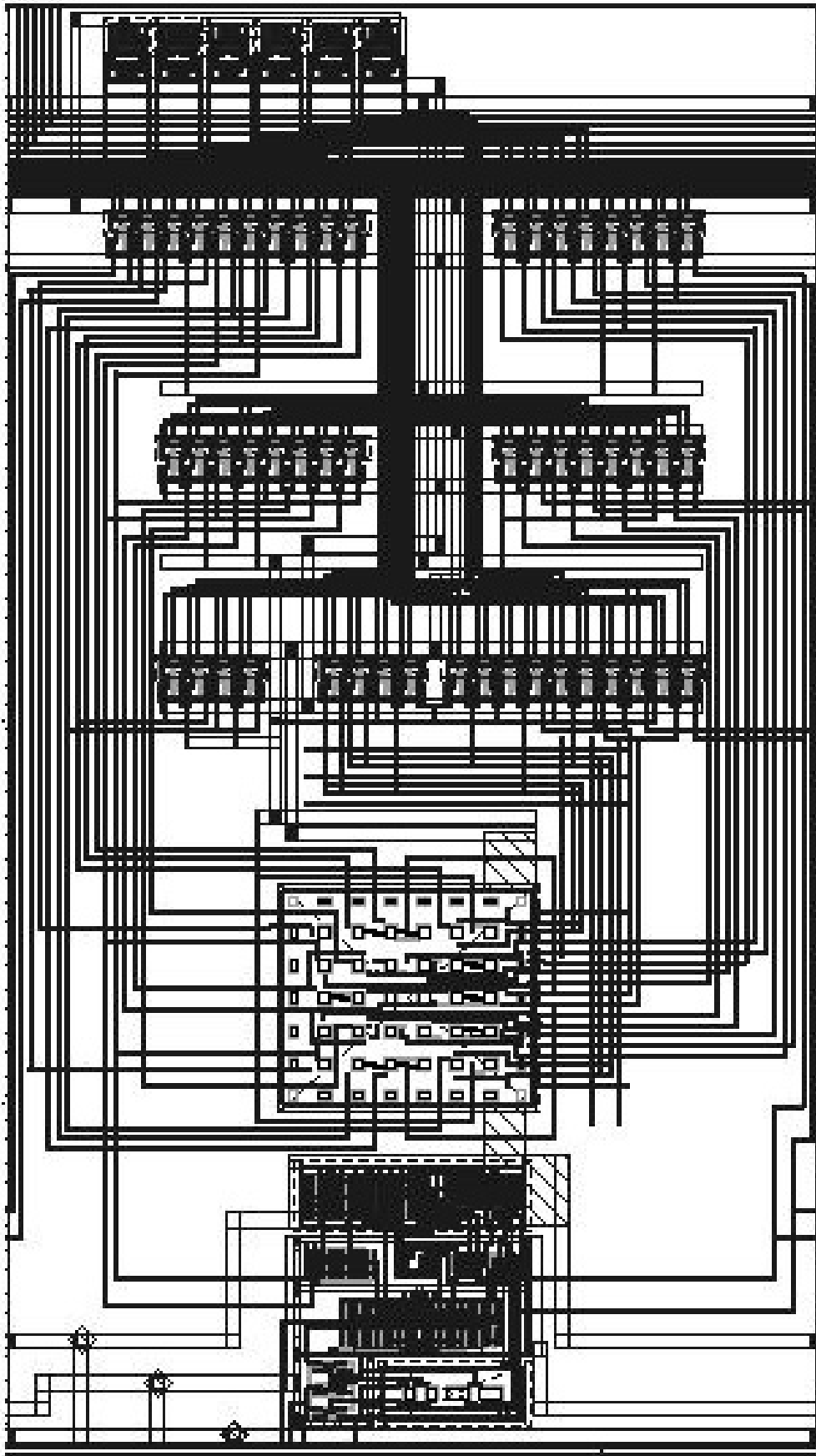


Fig. 5.13: *Layout*: módulo de zeros.

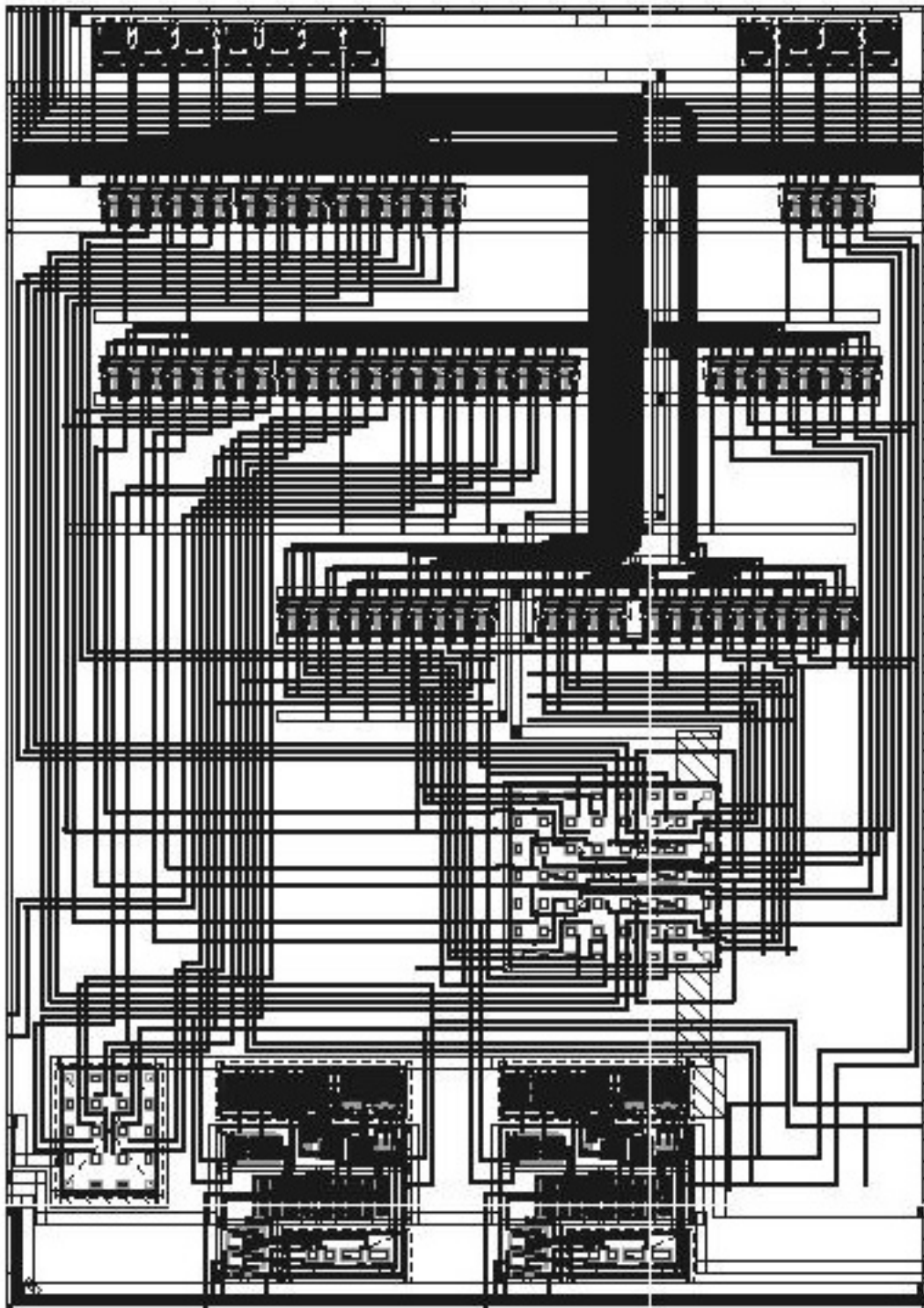


Fig. 5.14: *Layout*: módulo de pólos.

5.7. Implementação

Para a fabricação do circuito integrado foi utilizado o processo AMS CYE 0.8 μ m, com encapsulamento cerâmico de 68 pinos conforme mostrado na Fig. 5.15. A descrição dos pinos está na Tabela 5.3.

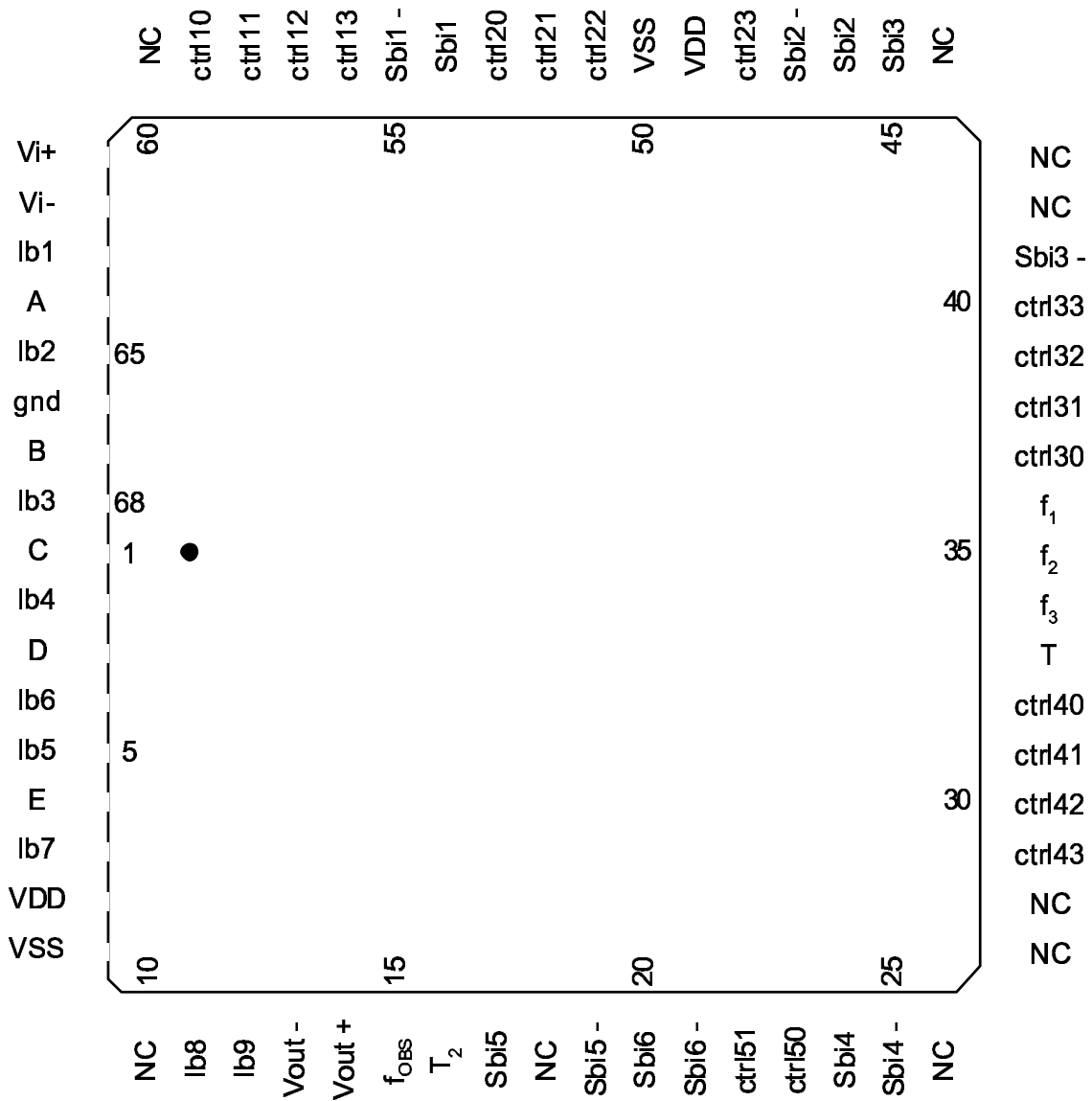


Fig. 5.15: Circuito integrado: encapsulamento cerâmico 68 pinos.

Tabela 5.3: Identificação dos pinos do circuito integrado

PINO		DESCRIÇÃO
1	C	Fase de controle do módulo seletor 3
2	Ib4	Corrente de polarização do amp-op do módulo de zeros 4
3	D	Fase de controle do módulo seletor 4
4	Ib6	Corrente de polarização do amp-op (somador) do módulo de pólos
5	Ib5	Corrente de polarização do amp-op do módulo de pólos
6	E	Fase de controle do módulo seletor 5
7	Ib7	Corrente de polarização do amp-op do módulo <i>buffer</i> de saída (OP3)
8	VDD	Tensão de alimentação (+)
9	VSS	Tensão de alimentação (-)
10	NC	Não conectado
11	Ib8	Corrente de polarização do amp-op do módulo <i>buffer</i> de saída (OP4)
12	Ib9	Corrente de polarização do amp-op do módulo <i>buffer</i> de saída (OP1/2)
13	Vout-	Saída (-) do filtro
14	Vout+	Saída (+) do filtro
15	f _{OBS}	Fase de controle do módulo <i>buffer</i> de saída
16	T ₂	Fase T ₂
17	Sbi5	Fase de programação de α_p – Módulo de pólos
18	NC	Não conectado
19	Sbi5 -	Fase de programação de α_p , invertida – Módulo de pólos
20	Sbi6	Fase de programação de β – Módulo de pólos
21	Sbi6 -	Fase de programação de β , invertida – Módulo de pólos
22	ctrl51	Controle do banco de capacitores de α_p (<i>bit</i> 1) – Módulo de pólos
23	ctrl50	Controle do banco de capacitores de α_p (<i>bit</i> 0) – Módulo de pólos
24	Sbi4	Fase de programação de α_z – módulo de zeros 4
25	Sbi4 -	Fase de programação de α_z – módulo de zeros 4, invertida
26	NC	Não conectado
27	NC	Não conectado
28	NC	Não conectado
29	ctrl43	Controle de atenuação (<i>bit</i> 1) – módulo de zeros 4
31	ctrl42	Controle de atenuação (<i>bit</i> 0) – módulo de zeros 4
31	ctrl41	Controle do banco de capacitores de α_z (<i>bit</i> 1) – módulo de zeros 4
32	ctrl40	Controle do banco de capacitores de α_z (<i>bit</i> 0) – módulo de zeros 4
33	T	Fase T
34	f3	Fase 3

PINO		DESCRIÇÃO
35	f2	Fase 2
36	f1	Fase 1
37	ctrl30	Controle do banco de capacitores de α_z (<i>bit</i> 0) – módulo de zeros 3
38	ctrl31	Controle do banco de capacitores de α_z (<i>bit</i> 1) – módulo de zeros 3
39	ctrl32	Controle de atenuação (<i>bit</i> 0) – módulo de zeros 3
40	ctrl33	Controle de atenuação (<i>bit</i> 1) – módulo de zeros 3
41	Sbi3 -	Fase de programação de α_z – módulo de zeros 3, invertida
42	NC	Não conectado
43	NC	Não conectado
44	NC	Não conectado
45	Sbi3	Fase de programação de α_z – módulo de zeros 3
46	Sbi2	Fase de programação de α_z – módulo de zeros 2
47	Sbi2 -	Fase de programação de α_z – módulo de zeros 2, invertida
48	ctrl23	Controle de atenuação (<i>bit</i> 1) – módulo de zeros 2
49	VDD	Tensão de alimentação (+)
50	VSS	Tensão de alimentação (-)
51	ctrl22	Controle de atenuação (<i>bit</i> 0) – módulo de zeros 2
52	ctrl21	Controle do banco de capacitores de α_z (<i>bit</i> 1) – módulo de zeros 2
53	ctrl20	Controle do banco de capacitores de α_z (<i>bit</i> 0) – módulo de zeros 2
54	Sbi1	Fase de programação de α_z – módulo de zeros 1
55	Sbi1 -	Fase de programação de α_z – módulo de zeros 1, invertida
56	ctrl13	Controle de atenuação (<i>bit</i> 1) – módulo de zeros 1
57	ctrl12	Controle de atenuação (<i>bit</i> 0) – módulo de zeros 1
58	ctrl11	Controle do banco de capacitores de α_z (<i>bit</i> 1) – módulo de zeros 1
59	ctrl10	Controle do banco de capacitores de α_z (<i>bit</i> 0) – módulo de zeros 1
60	NC	Não conectado
61	Vi+	Não conectado
62	Vi-	Não conectado
63	Ib1	Corrente de polarização do amp-op do módulo de zeros 1
64	A	Fase de controle do módulo seletor 1
65	Ib2	Corrente de polarização do amp-op do módulo de zeros 2
66	gnd	terra
67	B	Fase de controle do módulo seletor 2
68	Ib3	Corrente de polarização do amp-op do módulo de zeros 3

Fotografias mostrando detalhes de alguns módulos, obtidas com microscópio ótico de reflexão, são apresentadas nas Figs. 5.16 até 5.19.

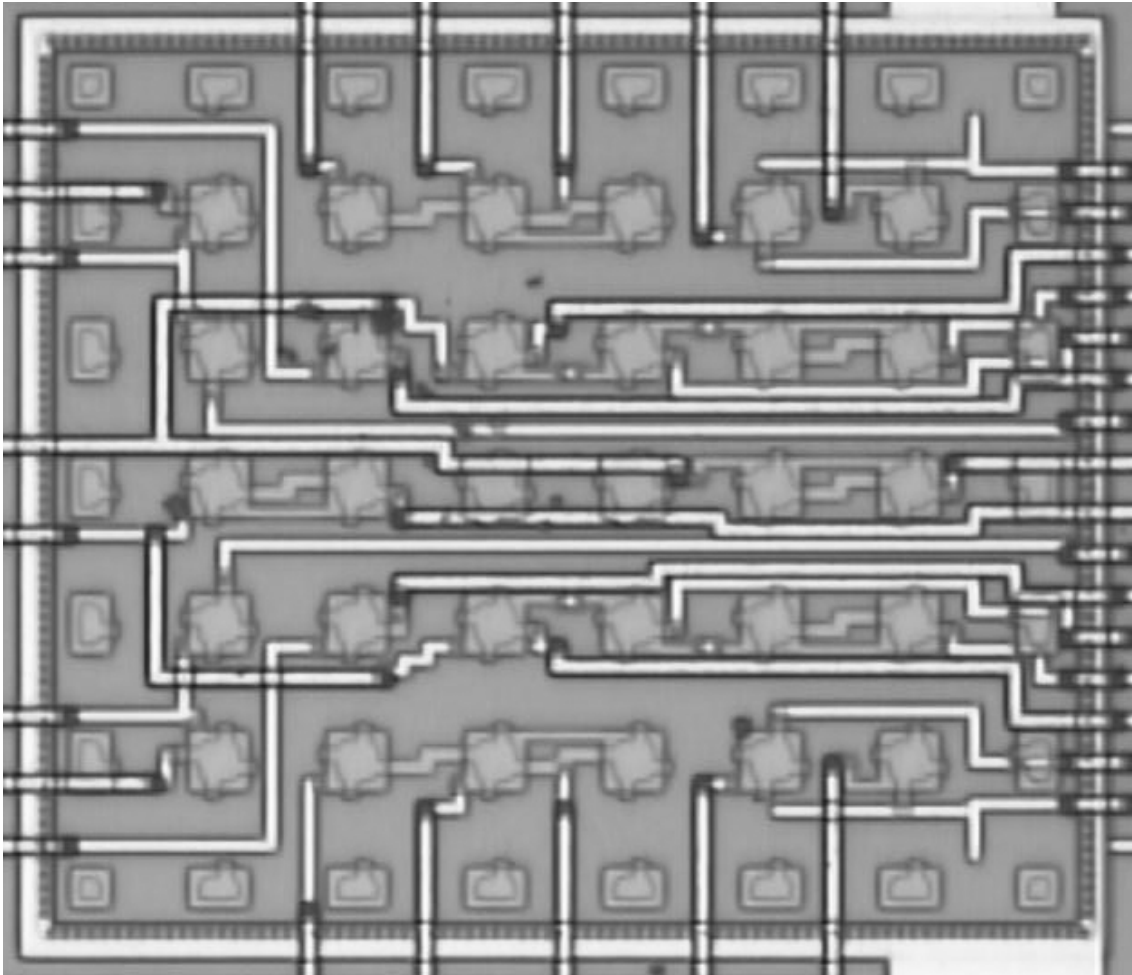


Fig. 5.16: Foto do banco de capacitores do módulo de zeros.

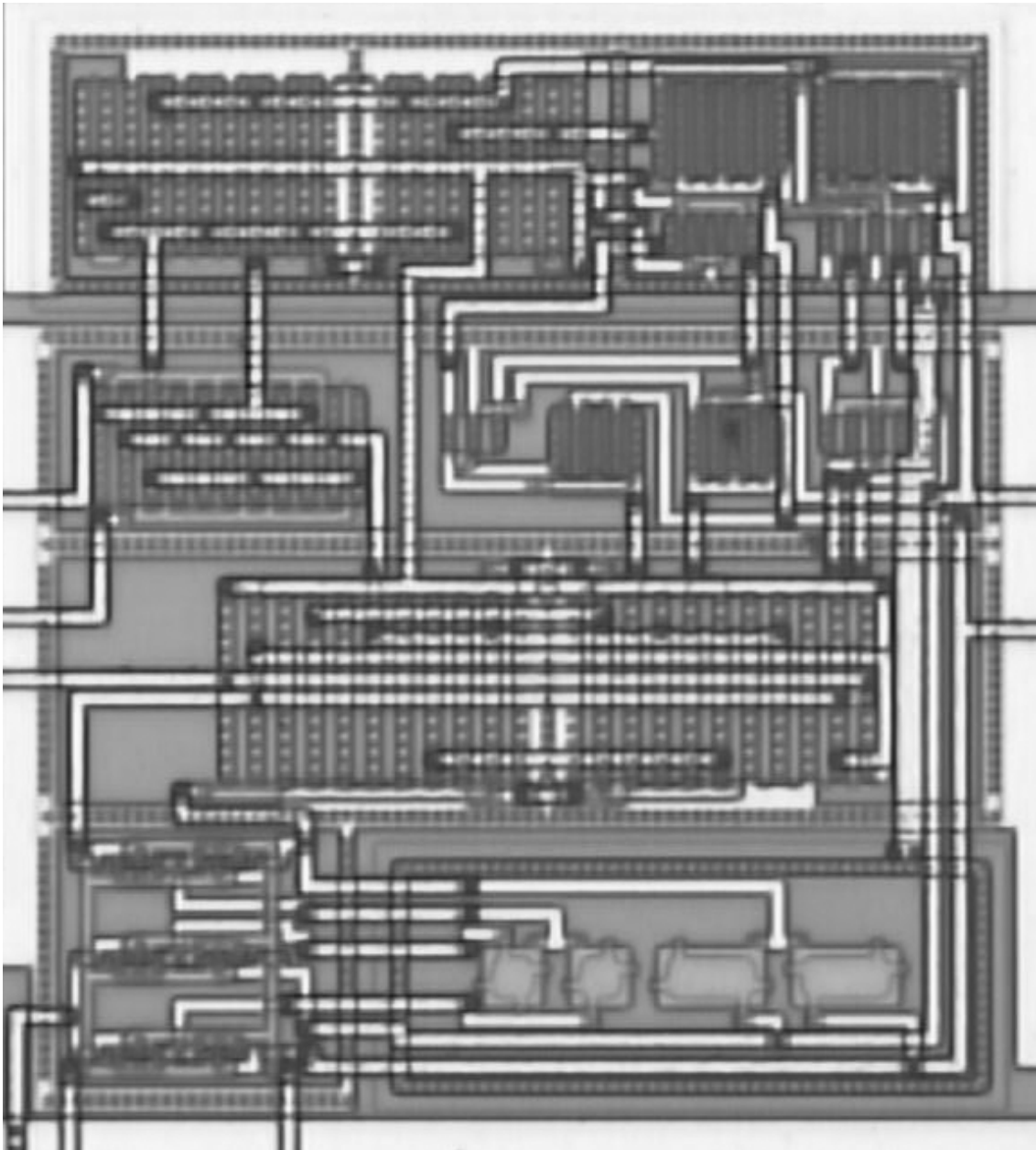


Fig. 5.17: Foto do amplificador operacional.

Na camada superior de cada módulo de zeros (Fig. 5.18) ou de pólos (Fig. 5.19), estão os inversores responsáveis pela geração das fases complementares necessárias no acionamento das chaves, que estão localizadas nas três camadas seguintes. Nas duas camadas restantes, estão o banco de capacitores e o(s) amplificador(es) operacional(is). Deve-se observar que as linhas de alimentação e de sinais digitais para o acionamento das chaves são distribuídas verticalmente pelo centro, evitando o cruzamento com as linhas laterais que fazem a conexão das chaves com o banco de capacitores e o(s) amp-op(s).

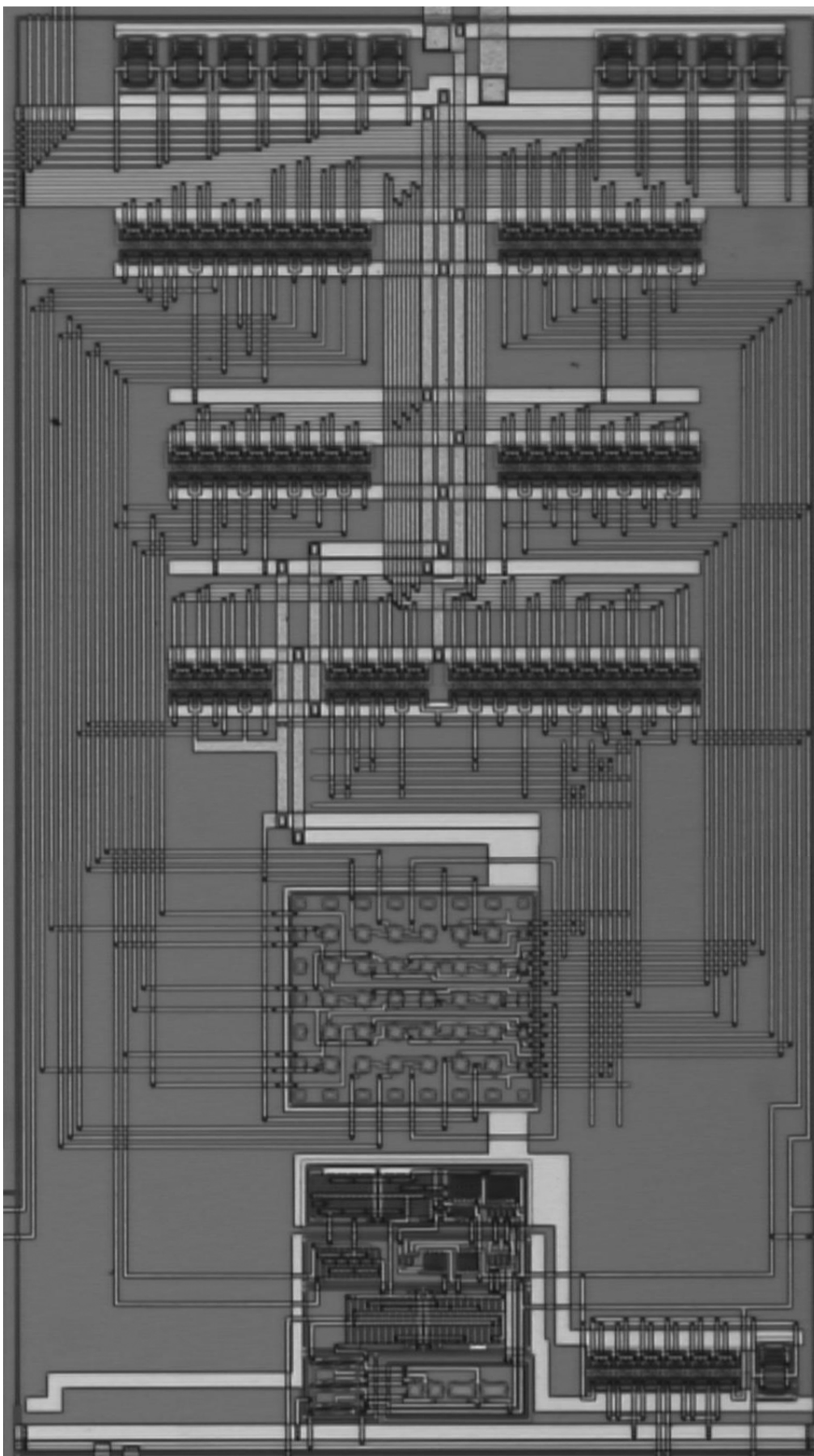


Fig. 5.18: Foto do módulo de zeros.

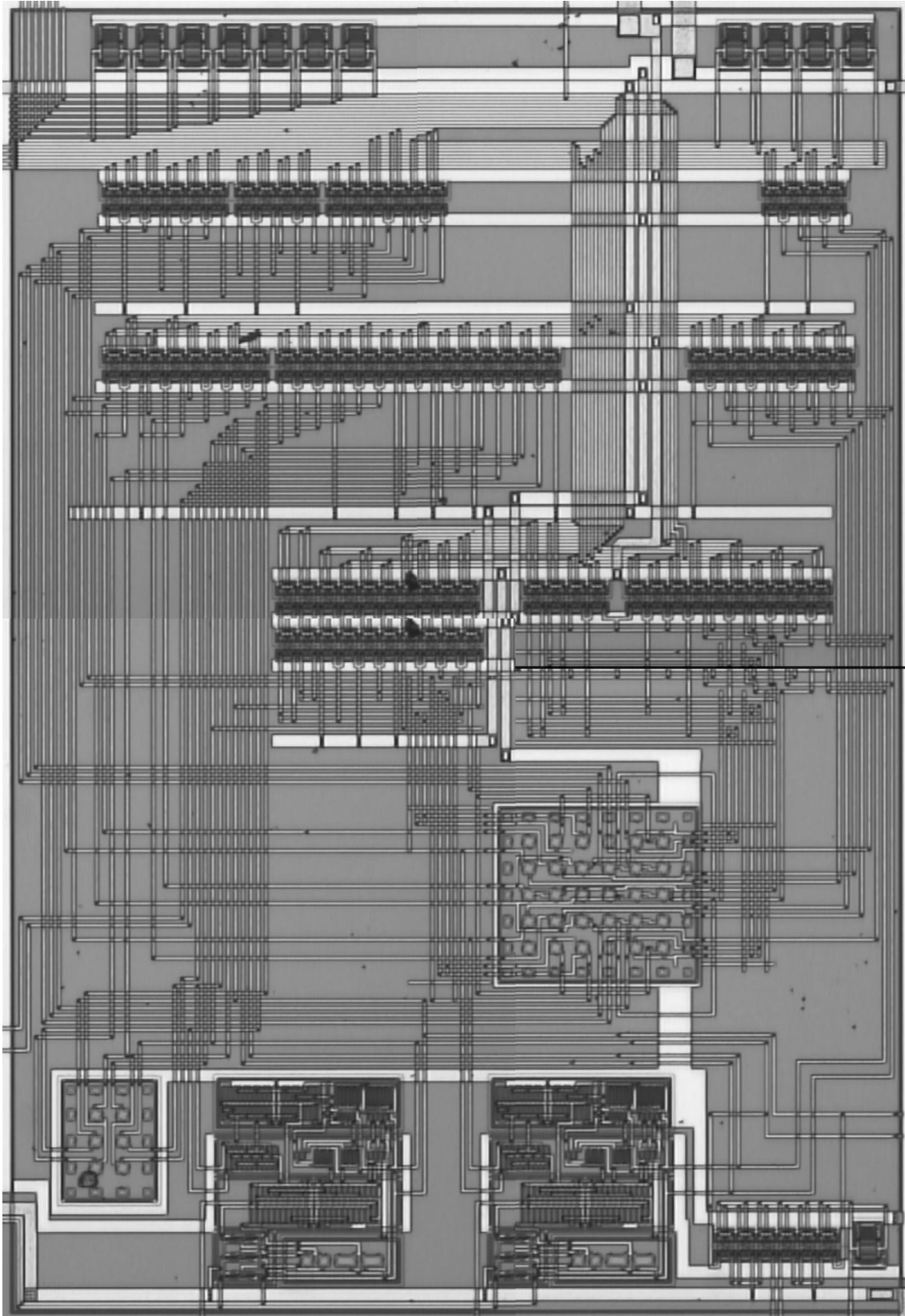
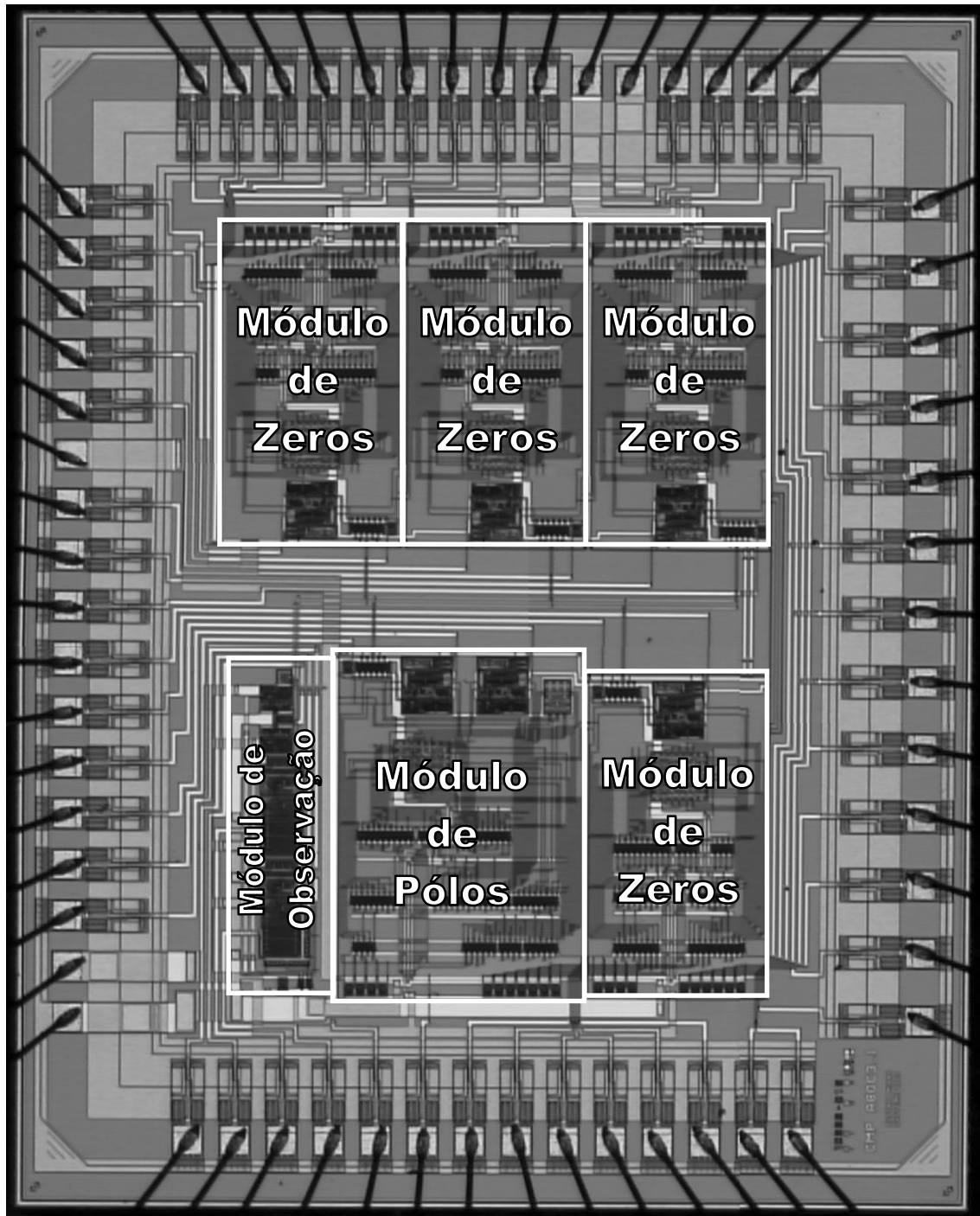
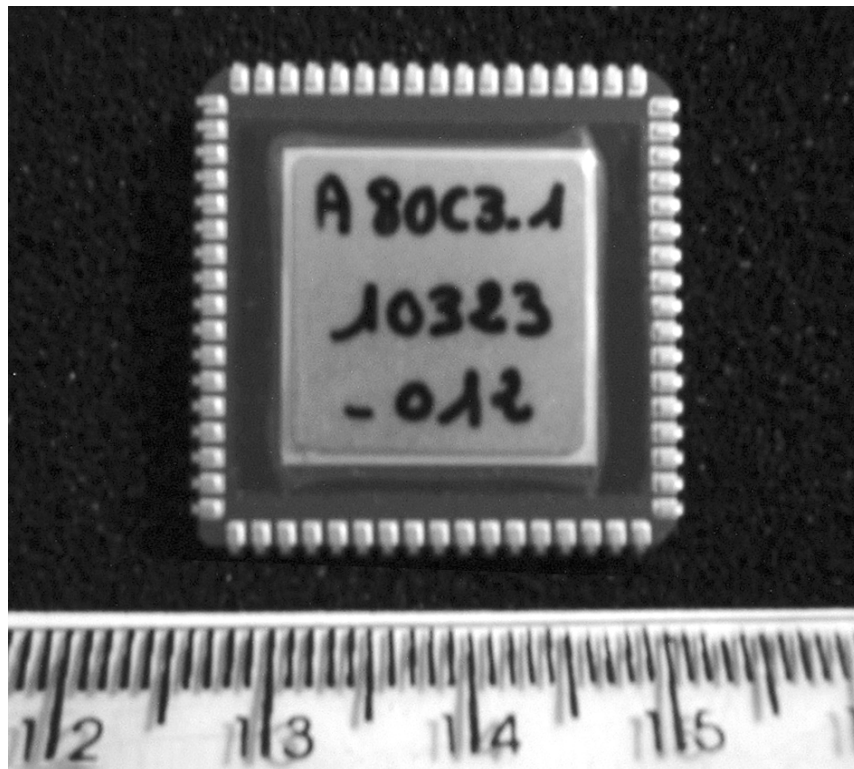


Fig. 5.19: Foto do módulo de pólos.

A Fig. 5.20 mostra as fotos interna e externa do circuito integrado completo. Na camada superior e à direita da camada inferior, estão localizados os quatro módulos de zeros. Na camada inferior, no centro e à esquerda, estão localizados, respectivamente, o módulo de pólos e o módulo *buffer* de saída.



(a)



(b)

Fig. 5.20: Fotografia do circuito integrado: (a) interna; (b) aspecto externo

5.8. Placa de Teste

Foi criada uma placa de teste dedicada, em circuito impresso, contendo todos os recursos necessários à operação do circuito integrado, sendo responsável pela geração de todos os sinais digitais utilizados. Ela incorpora o estabilizador de tensão para a alimentação do circuito; o cristal oscilador; o gerador das fases de teste; as fontes de corrente para a polarização dos amp-ops; as chaves DIP para programação do filtro e os circuitos de conversão do sinal de unifilar para diferencial, na entrada, e diferencial para unifilar, na saída, usando amplificador operacional comercial de alto GB. Os sinais gerados são doze fases programáveis (Sbi1 a Sbi6 e suas complementares) e cinco fases fixas (f_1 , f_2 , f_3 , T e T_2). As fases f_1 , f_2 , f_3 e T são comuns a todos os módulos, a fase T_2 atua somente no módulo de pólos. As fases programáveis são duas para cada módulo de zeros e quatro para o módulo de pólos. Devido à sua complexidade, detalhes do projeto do gerador de fases e a forma de programação são apresentados no Apêndice A.

5.9. Programação

Para a programação do filtro deve-se adotar o seguinte procedimento:

- 1 – Projetar o filtro de estrutura fixa com 8 zeros e 2 pólos complexos, utilizando-se o programa de otimização `iirremez.m` [23], desenvolvido para o projeto de filtros com número de zeros diferente do número de pólos;
- 2 – Decompor o numerador da função de transferência em quatro polinômios de 2ª ordem. Identificar os coeficientes α_z e o ganho DC com que cada módulo de zeros irá contribuir;
- 3 – Ordenar os zeros de acordo com o seguinte critério: primeiro, em ordem crecente, o(s) zero(s) com coeficiente(s) α_z nulo(s) ou positivo(s). Em seguida, em ordem decrescente, o(s) o(s) zero(s) com coeficiente(s) negativo(s).
- 4 – Ajustar a faixa dinâmica do filtro, indicando a atenuação adequada (1, 1/2, 1/3 ou 1/4) para cada módulo de zeros, de modo que o ganho total do módulo seja igual ou inferior a 1;
- 5 – Identificar β e calcular α_p do módulo de pólos;
- 6 – Verificar nas tabelas de programação (Tabela B.1 e Tabela B.2) o coeficiente que mais se aproxima de cada um dos coeficientes α_z , α_p e β , com os respectivos códigos binários;
- 7 – Programar a placa de teste e operar o sistema.

Para facilitar a aplicação da seqüência de procedimentos de 1 a 6, foi desenvolvido no Matlab, o programa `Calc_coef_filtro.m` que indica todos os códigos binários necessários à programação da placa de teste. O código fonte do programa é apresentado no Apêndice B.

Capítulo 6

Resultados Experimentais

Neste capítulo são apresentados os resultados dos testes realizados em laboratório com o circuito integrado fabricado.

6.1. Consumo

Como os inversores só consomem corrente nas transições das fases, o consumo de potência destes componentes depende dos tempos de subida e descida das fases. Como temos fases com frequências diferentes e a maioria programáveis, cada inversor terá o seu consumo próprio de difícil avaliação. Optou-se por caracterizar o consumo em três partes: o consumo residual, referente aos inversores; o consumo dos amp-ops projetados e o consumo do módulo buffer de saída. Deve-se considerar como consumo do circuito integrado, apenas a soma do consumo residual com o consumo dos amp-ops, uma vez que o módulo de saída existe somente para que se possa observar o sinal processado.

Primeiramente foi medido o consumo residual com todas as fontes de corrente dos amp-ops desligadas. Posteriormente, o consumo foi novamente medido para cada amp-op polarizado ($I_b = 50\mu A \Leftrightarrow I_o = 200\mu A$) e, por diferença, obteve-se o consumo de cada amp-op. Finalmente, o consumo do módulo de saída foi medido polarizando-se os seus amplificadores e desligando-se todos os outros amp-ops dos módulos de zeros e pólos.

Tabela 6.1: Consumo de potência.

	CONSUMO DE POTÊNCIA (mW)		
	teórico	simulado	experimental
residual (inversores)	–	–	3,80
amp-op	4,75	4,58	4,66
filtro (residual + 6 amp-ops)	–	–	31,76
buffer de saída	–	46,40	43,40

Na Tabela 6.1 são apresentados os resultados das medidas podendo-se concluir que os espelhos de corrente dos amp-ops ficaram bastante precisos, uma vez que os consumos teórico e experimental apresentam um erro mínimo.

6.2. Distorção Harmônica

A medida da distorção harmônica total (THD) foi feita com o analisador de espectro disponível Hewlet Packard modelo HP3582A, com operação na faixa de áudio (até 25kHz). Como a frequência de amostragem (f_s) é 21 vezes menor que a frequência de *clock* (f_{ck}), foi necessário reduzir f_{ck} para 1MHz de modo a limitar a máxima frequência de operação do filtro ($f_s/2$) em 23809,5 Hz, permitindo a observação de suas características com o instrumento utilizado.

Para esta medição, o circuito integrado foi programado para a resposta do Filtro 1 (Apêndice B). Para o sinal aplicado escolheu-se a frequência de 1kHz que, havendo distorção, pode apresentar pelo menos seis harmônicos dentro da banda passante do filtro. Na Tabela 6.2 e na Fig. 6.1 são apresentados, respectivamente, os resultados das medidas de THD e as formas de onda do sinal, normal e com média no tempo, com amplitude de $1V_p$ (0,99% THD), obtidas com o osciloscópio Tektronix modelo TDS360.

Tabela 6.2: Distorção harmônica

DISTORÇÃO HARMÔNICA TOTAL (THD)		
Nível de saída (V)	THD (%)	THD (dB)
0,5	0,18	-54,9
1,01	0,99	-40,1
1,1	3,79	-28,4
1,2	6,46	-23,8

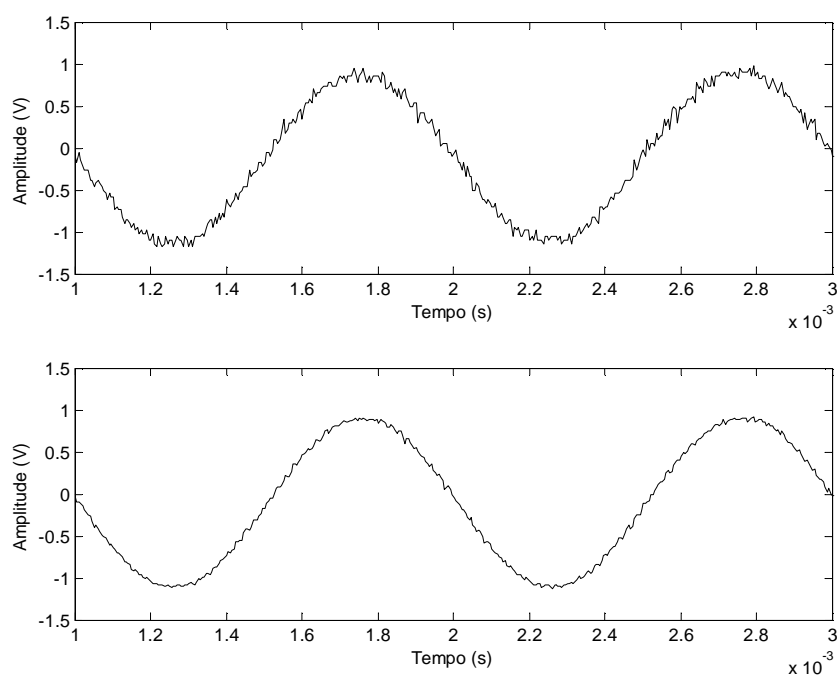


Fig. 6.1: Sinal de saída com 0,99% de THD: normal e com média no tempo.

6.3. Faixa Dinâmica

A faixa dinâmica é definida como a razão entre a máxima amplitude RMS do sinal de saída com 1% de THD e o ruído total na banda de passagem. A medida da densidade espectral de ruído foi realizada, primeiramente, considerando apenas o dispositivo de teste e, posteriormente, incluindo o filtro em análise. Para o dispositivo de teste obteve-se um valor aproximadamente constante e igual a $11,40\mu\text{V}/\sqrt{\text{Hz}}$. Para o conjunto o resultado está indicado na Fig. 6.2.

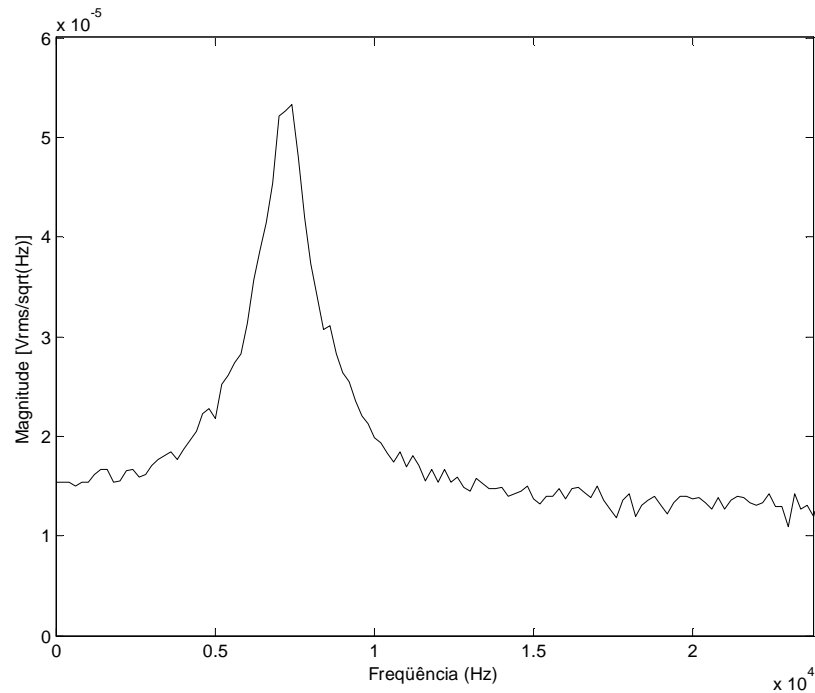


Fig. 6.2: Densidade espectral de ruído ($V_{\text{rms}} / \sqrt{\text{Hz}}$)

Retirando a influência do ruído do dispositivo de teste, pode-se calcular o ruído total na banda de passagem do filtro em teste (7,1kHz), em aproximadamente 1,61mVrms. Assim, a faixa dinâmica resultante é de 52,8dB. Vale lembrar que optou-se pelo uso de capacitores unitários de 100fF devido à experiência anterior bem sucedida [20]. Entretanto, sob o ponto de vista da redução de ruído, seria interessante o aumento destes elementos, uma vez que, sendo o ruído térmico inversamente proporcional ao valor do capacitor utilizado, resultaria em um filtro com faixa dinâmica maior.

6.4. Programabilidade

Para verificar a programabilidade do filtro foram realizadas, inicialmente, medidas da resposta em frequência do primeiro módulo de zeros, cujas saídas foram acessadas pelo acionamento da chave seletora correspondente (ver Fig. 5.6). Com palavras de

8 bits o coeficiente α_z pode ser programado em 256 passos. O ajuste grosso do coeficiente, obtido com os 2 bits de programação do banco de capacitores, permite o posicionamento do zero em uma das quatro faixas (ver Apêndice B) em que pode ser dividida a semicircunferência de raio unitário. Os 6 bits restantes, usados na operação do DAC, controlam o ajuste fino, permitindo o posicionamento do zero em qualquer uma das 64 posições possíveis dentro de cada faixa de ajuste grosso.

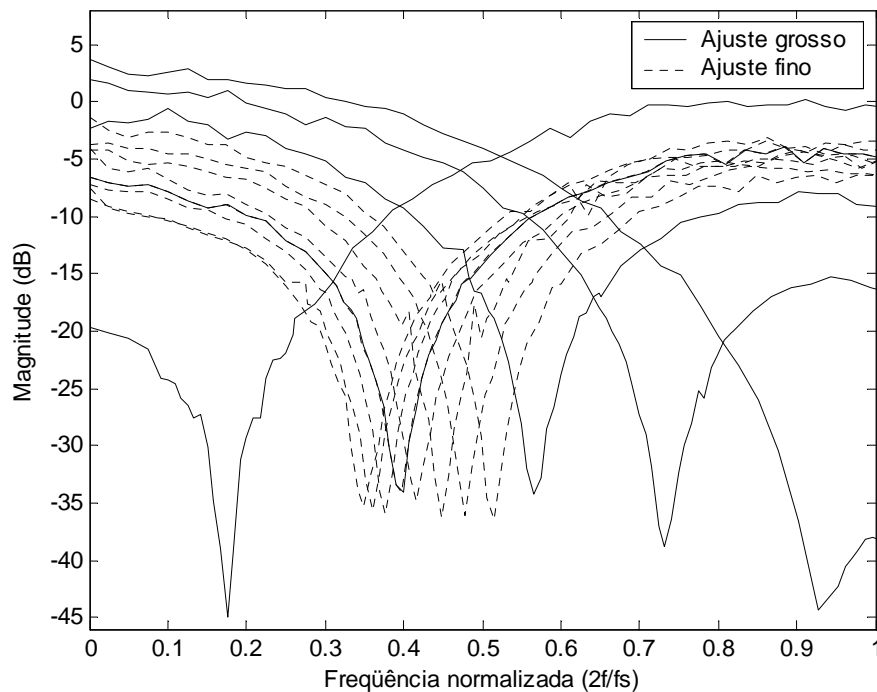
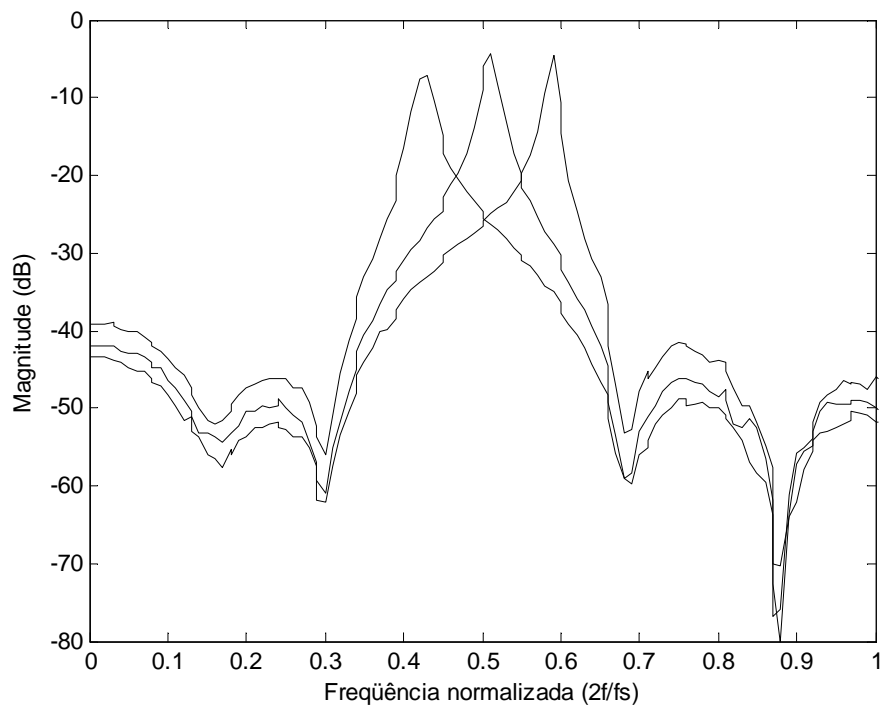


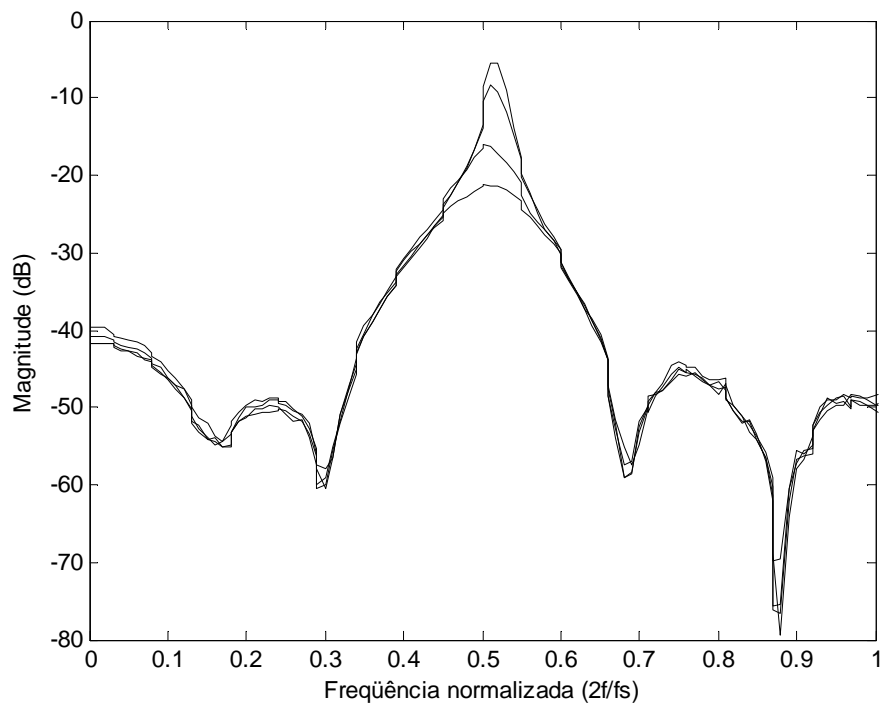
Fig. 6.3: Programabilidade do módulo de zeros

Na Fig. 6.3 são mostrados os resultados das medidas das respostas em frequência, em termos da frequência normalizada $2f/f_s$, para várias programações do coeficiente α_z . Para maior clareza do gráfico, o ajuste fino é mostrado apenas num pequeno intervalo, embora esta facilidade esteja disponível em toda a faixa de operação do filtro.

Na Fig. 6.4. estão apresentados os resultados dos testes de programabilidade dos pólos.



(a)



(b)

Fig. 6.4: Programabilidade do módulo de pólos: (a) ajuste de α_p ; (b) ajuste de β .

Como o módulo de pólos é posicionado após os módulos dos zeros na estrutura do filtro, só é possível observar a sua resposta acompanhada das respostas de todos os zeros. Para observar a sua programabilidade optou-se por sintonizar a freqüência do pólo no centro da faixa com dois zeros de cada lado numa configuração de filtro seletivo. Para verificar a influência do ajuste de α_p na freqüência de sintonia do pólo, o coeficiente β foi mantido no seu valor máximo, enquanto foram aplicados três ajustes em α_p , variando as freqüências de sintonia de 0,42 a 0,58.

Para verificar a influência do ajuste de β na seletividade do pólo, o coeficiente α_p foi mantido constante, posicionando a freqüência de sintonia do pólo no centro da banda, enquanto β foi variado de 0,70 a 0,98.

6.5. Resposta em Freqüência

Para testar o funcionamento global da estrutura proposta, foram projetados filtros com 8 zeros e 2 pólos complexos utilizando-se o programa de otimização `iirremez.m` [23], desenvolvido para o projeto de filtros com número de zeros diferente do número de pólos. As características dos filtros projetados estão relacionadas na Tabela 6.3.

Tabela 6.3: Características dos filtros projetados

Características	Filtro 1	Filtro 2	Filtro 3	Filtro 4
<i>ripple</i> na banda de passagem (dB)	1,0	3,0	1,0	3,0
atenuação na banda de rejeição (dB)	52	23	25	35
freqüência de corte normalizada	0,30	0,27	0,15	0,30

Seguindo o procedimento descrito na seção 5.9, foram gerados os códigos binários para a programação dos filtros. As tabelas são apresentadas no Apêndice B.

Nas Figs. 6.5 até 6.8 são apresentadas, respectivamente, as comparações entre as respostas em frequência experimentais e teóricas dos quatro filtros projetados. As curvas teóricas já estão distorcidas simulando o efeito do *sample and hold* na saída do filtro.

Deve-se observar que as curvas teórica e experimental estão bem próximas, indicando que as técnicas utilizadas para minimizar as influências de capacitâncias parasitas foram eficazes para a realização dos ajustes.

É importante observar, também, que além da precisão em frequência, a grande atenuação e o aspecto da curva apresentada pelos zeros indicam que eles estão localizados muito próximos da circunferência de raio unitário, e sinalizam o correto funcionamento da estrutura proposta com este objetivo.

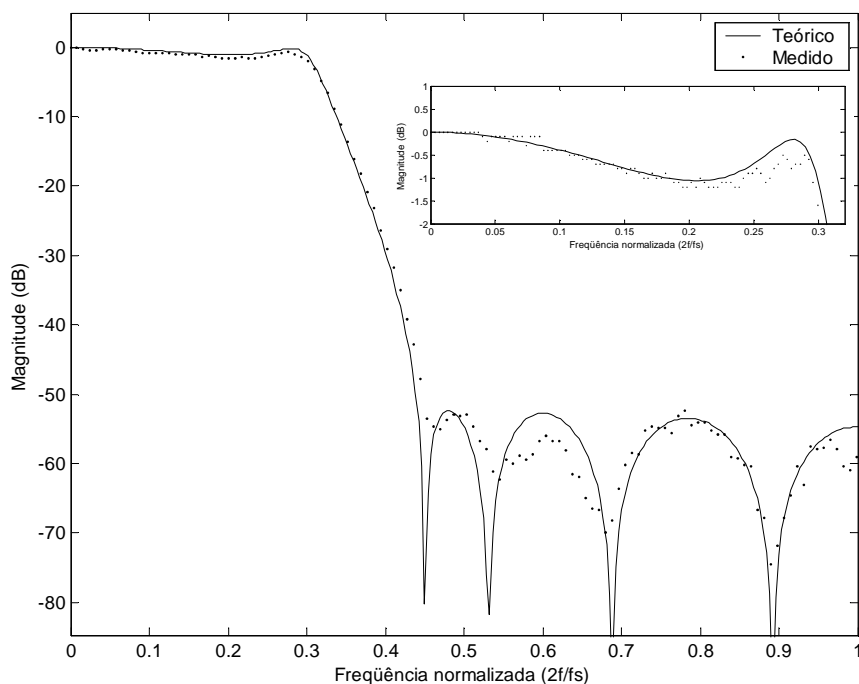


Fig. 6.5: Resposta em frequência do filtro 1 com detalhe da banda passante.

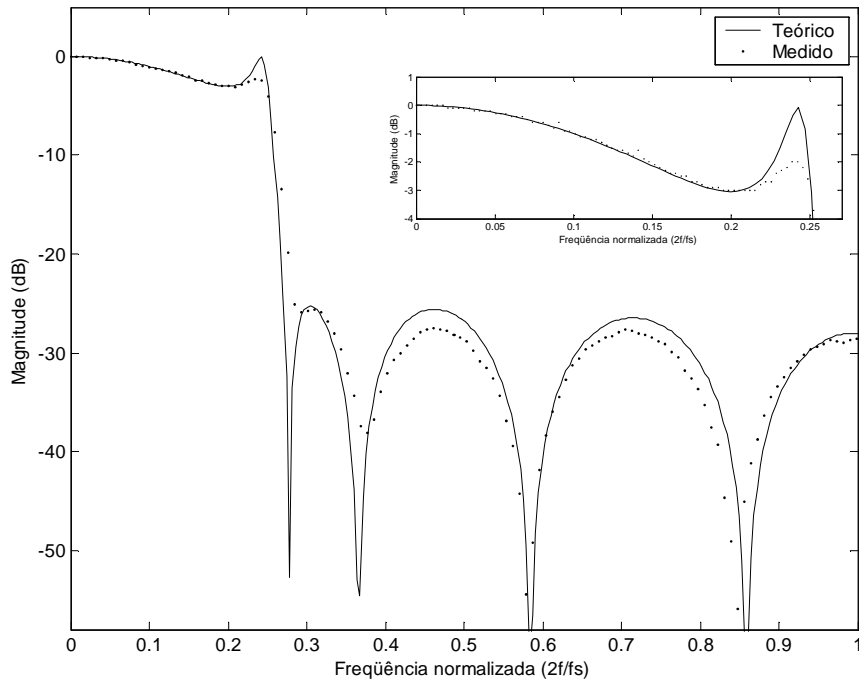


Fig. 6.6: Resposta em frequência do filtro 2 com detalhe da banda passante.

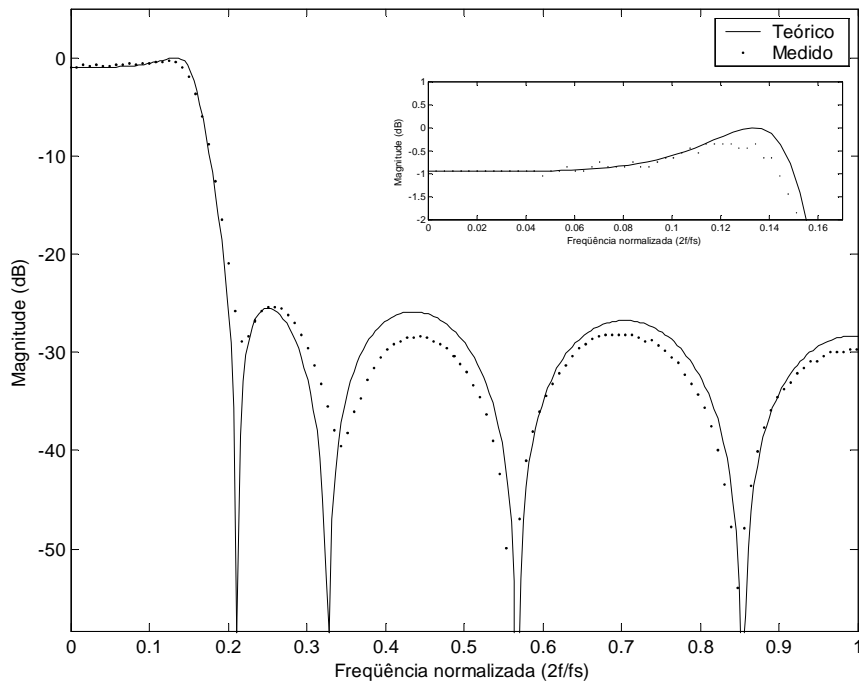


Fig. 6.7: Resposta em frequência do filtro 3 com detalhe da banda passante.

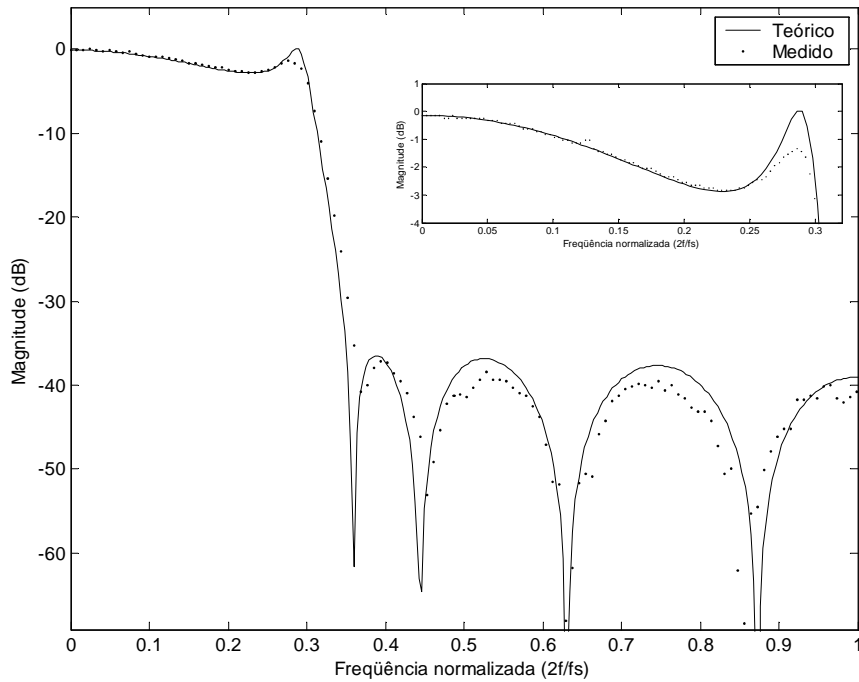
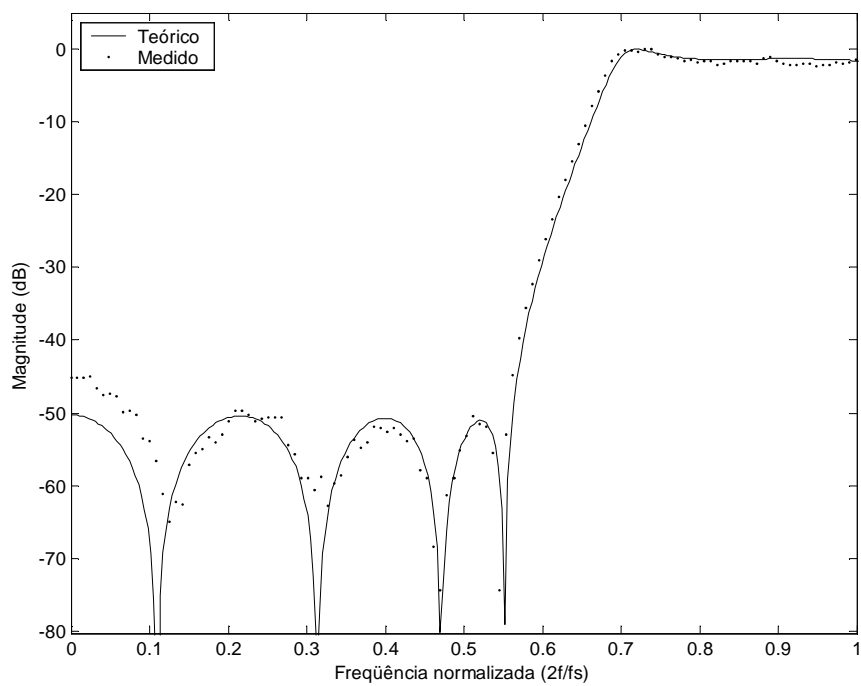
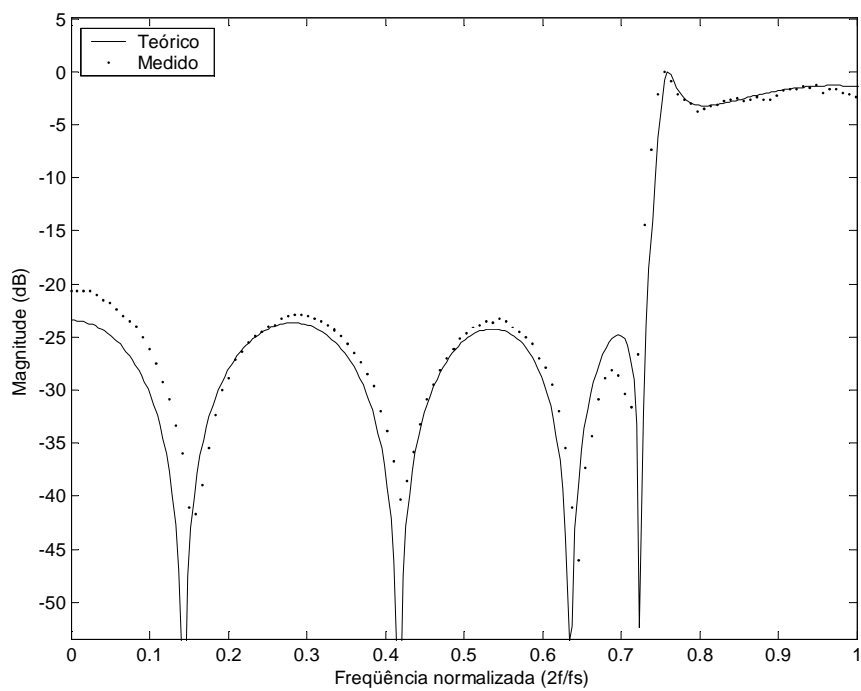


Fig. 6.8: Resposta em frequência do filtro 4 com detalhe da banda passante.

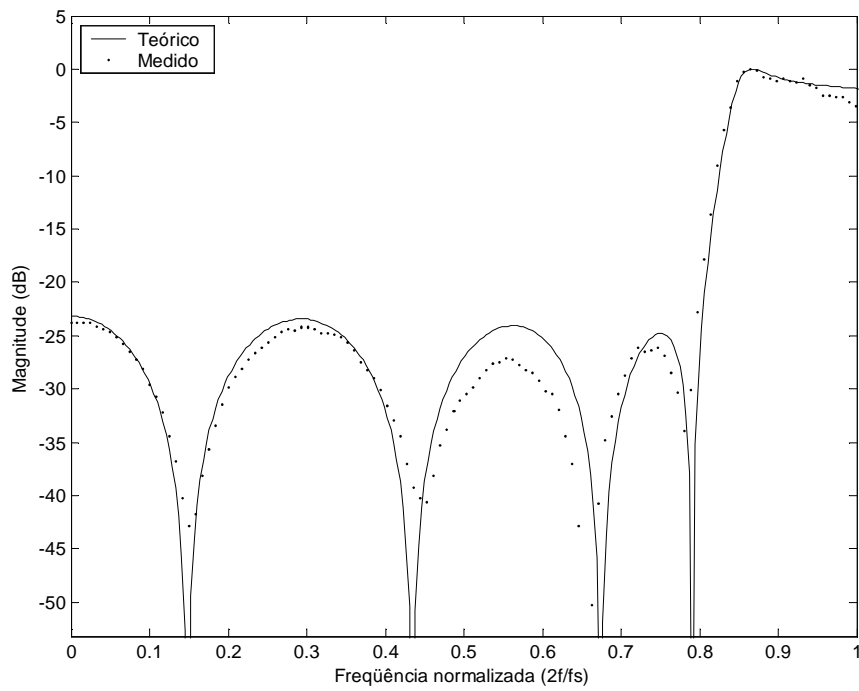
Como ilustração da versatilidade da estrutura proposta, são apresentadas na Fig. 6.9 as respostas em frequência de filtros passa-altas baseados no espelhamento (transformação $z \rightarrow -z$) do posicionamento de pólos e zeros dos quatro filtros passa-baixas já apresentados. Deve-se notar que os pequenos desvios observados no primeiro zero das respostas experimentais apresentadas nas Figs. 6.9 (a), (b) e (d), e no pólo da Fig. 6.9 (c) podem ser facilmente corrigidos pela alteração da programação. Um pequeno deslocamento para a esquerda dos zeros e do pólo mencionados é suficiente para corrigir os desvios das quatro curvas de resposta.



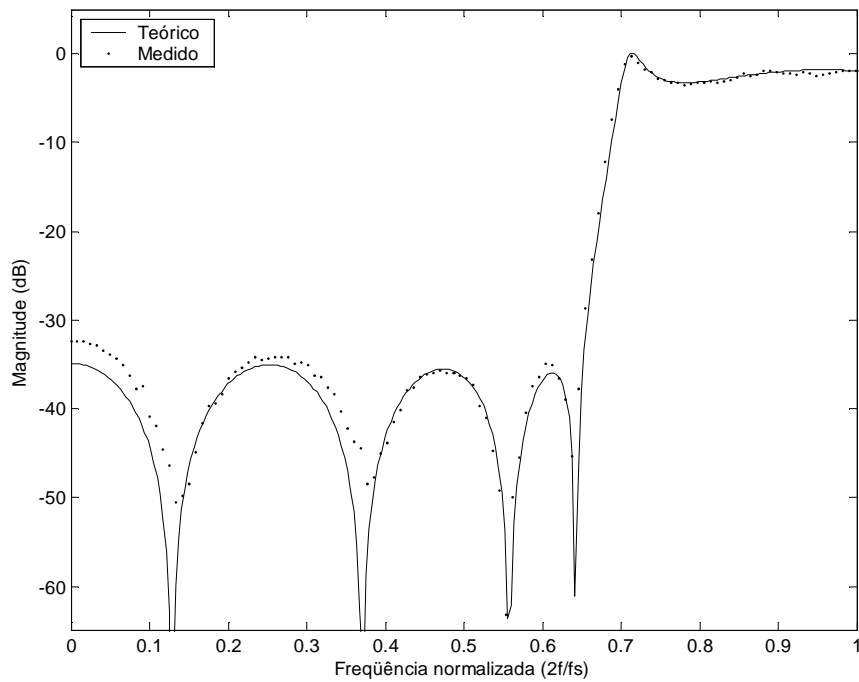
(a)



(b)



(c)

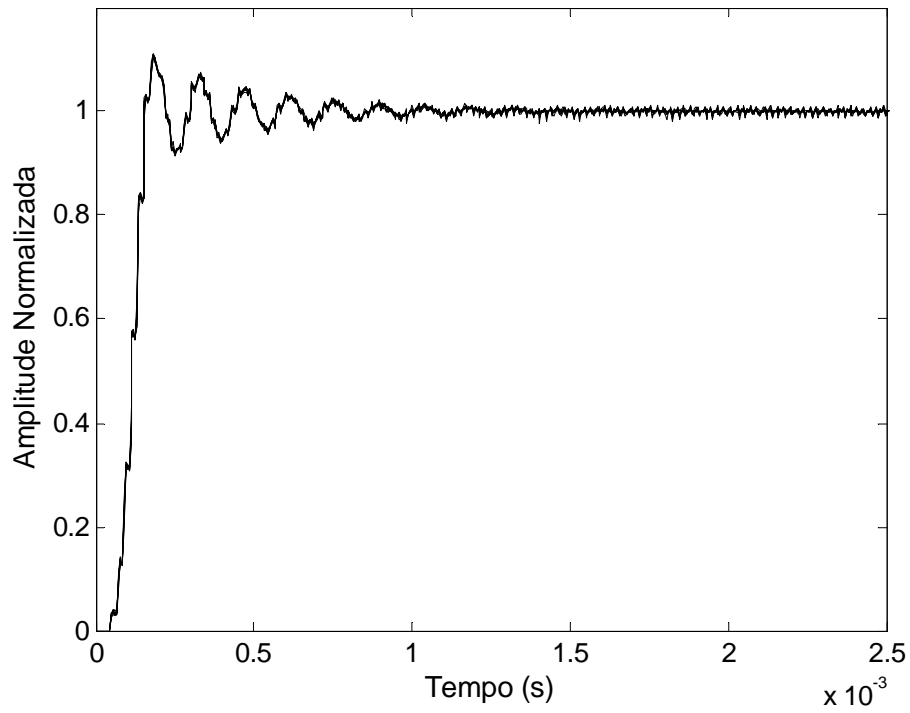


(d)

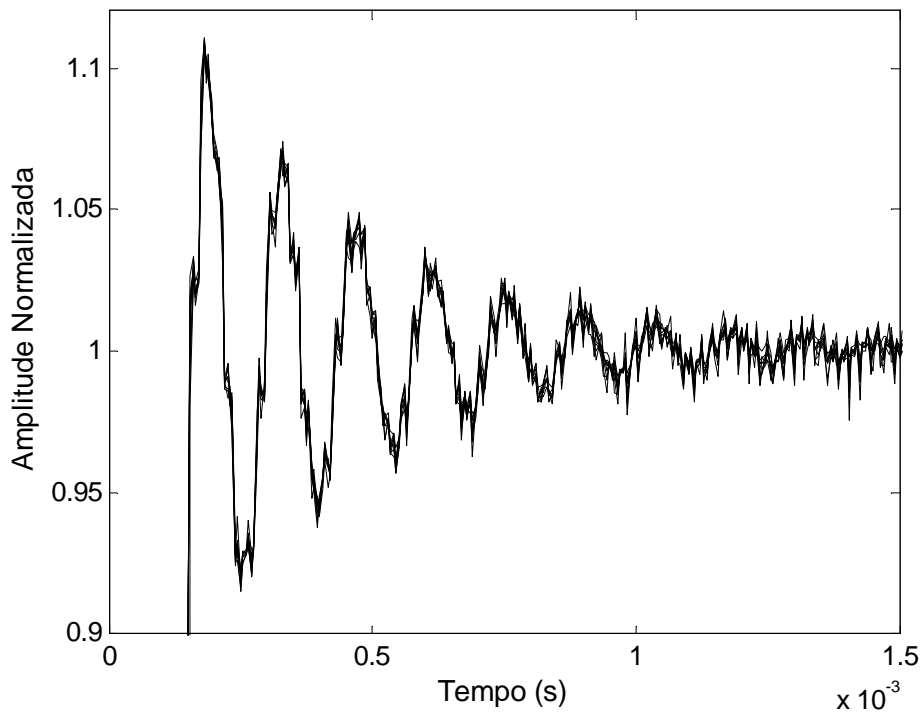
Fig. 6.9: Filtro passa-altas, gerados pelo espelhamento (transformação $z \rightarrow -z$) das estruturas passa-baixas anteriores: (a) filtro 1; (b) filtro 2; (c) filtro 3 e (d) filtro 4.

6.6. Uniformidade

Uma vez que o analisador de espectro utilizado não disponibiliza um arquivo com as medidas realizadas, o levantamento das curvas de resposta em frequência de todos os CIs disponíveis seria uma tarefa muito demorada. Entretanto, mesmo com poucas amostras (nove), é importante conhecer o comportamento do filtro projetado sob o aspecto de uniformidade em relação ao processo de fabricação. Como alternativa foi realizada a medida da resposta ao degrau, uma vez que o posicionamento de pólos e zeros influencia diretamente na resposta transiente do filtro e, desta forma, qualquer variação neste posicionamento pode ser observada pela alteração do *overshoot* e da frequência natural de oscilação. Para estas medidas foi escolhido o filtro 4 (Apêndice B) e, com a programação fixa, foram testados CIs disponíveis (nove) utilizando-se o osciloscópio Tektronix TDS360, que disponibiliza em meio magnético as formas de onda obtidas. Os resultados das medidas dos nove filtros foram traçados num único gráfico (Fig. 6.10), onde se pode constatar a uniformidade de comportamento dos CIs, indicando que as técnicas utilizadas para minimizar os efeitos do processo de fabricação podem garantir uma boa reprodutibilidade do circuito integrado.



(a)



(b)

Fig. 6.10: Uniformidade: (a) comparação da resposta ao degrau de nove CIs; (b) detalhe da resposta transiente.

Capítulo 7

Conclusões

Os resultados obtidos mostram que a estrutura apresentada pode ser utilizada com vantagens na realização de filtros analógicos digitalmente programáveis. O número de zeros maior do que o número de pólos garante a baixa sensibilidade à variação de coeficientes. O ajuste do capacitor pela técnica de conversão digital-analógica (DAC) permite a programação de maneira simples, e conduz a soluções com baixa dispersão dos valores dos capacitores que, aliada à modularidade da estrutura, são características atraentes para a integração tanto pela facilidade do projeto, como pela redução significativa da área de capacitores necessária para a implementação em circuito integrado. Quantitativamente, estima-se um aumento de 64 vezes na área de capacitores e 2,6 vezes no consumo, para o caso deste mesmo filtro programado por banco de capacitores ao invés da técnica de controle de cargas (DAC). O aumento da área se deve aos 256 capacitores unitários que seriam necessários para a realização de um banco de capacitores com resolução de 8 *bits* na posição de C_4 , mais o aumento correspondente dos demais capacitores para a implementação correta da função de transferência. O aumento do consumo é proporcionalmente menor, pois apesar do aumento da capacitância de carga, pode-se permitir um tempo maior para a estabilização do sinal nas saídas dos amplificadores operacionais, uma vez que o filtro operaria somente com as fases f_1 , f_2 e f_3 utilizadas nos módulos de pólos e zeros.

A reduzida quantidade de amp-ops, conseguida com a técnica de multiplexação, viabiliza a utilização da estrutura em aplicações onde haja exigência de baixo consumo de potência. Contudo, para operação com 8 *bits* de resolução, a frequência

de operação do DAC (f_{ck}) será 21 vezes maior do que a frequência de amostragem do filtro (f_s), impondo uma limitação quanto à sua frequência máxima de operação.

7.1. Trabalhos Futuros

Como trabalhos futuros destaca-se a criação de circuitos que implementem zeros genéricos de 1ª e 2ª ordens, programáveis, isto é, módulos que possibilitem a realização e posicionamento de zeros reais e complexos numa ampla região do plano complexo.

A elaboração de uma estrutura composta de vários destes módulos, além daqueles desenvolvidos neste trabalho, permitiria o desenvolvimento de um circuito integrado com facilidade de programação de coeficientes e, também, da estrutura do filtro. Este novo CI teria aplicação imediata em filtragem adaptativa para equalização de canais.

Apêndice A

Placa de Teste

Para a obtenção dos resultados experimentais foi desenvolvida uma placa de teste que incorpora todas as funções necessárias ao funcionamento do circuito integrado. O projeto inicial, baseado na utilização de portas lógicas discretas da série 74F de alta velocidade, necessitaria de mais de 60 CIs para a geração dos sinais de controle. Para reduzir o tamanho e complexidade da placa de circuito impresso necessária, optou-se por desenvolver um circuito baseado em tecnologia de FPGA (*Field Programmable Gate Array*), utilizando uma placa de avaliação da Spartan II XC2S100PQ208-5. O desenvolvimento do projeto lógico digital, cujo diagrama em blocos está indicado na Fig. A.1, foi realizado por meio de programação em linguagem específica (VHDL). Como as saídas da FPGA são de 3,3V e o circuito integrado do filtro precisa de 5V para o nível lógico “1” das fases de controle, houve necessidade de utilização de portas lógicas CMOS da série 74AC para compatibilizar os dois sistemas. A placa de teste projetada incorpora, ainda, o regulador de tensão, as fontes de corrente para polarização dos amp-ops, as chaves DIP para programação do filtro e o sistema de acoplamento do sinal com transformação unifilar para diferencial e vice-versa, utilizando o amplificador totalmente diferencial de alta velocidade THS4141 da Texas Instruments. O diagrama esquemático e a foto da placa são mostrados, respectivamente, nas Figs. A.2 até A.6. Em seguida estão relacionados os programas fonte, em linguagem VHDL, usados na programação da FPGA.

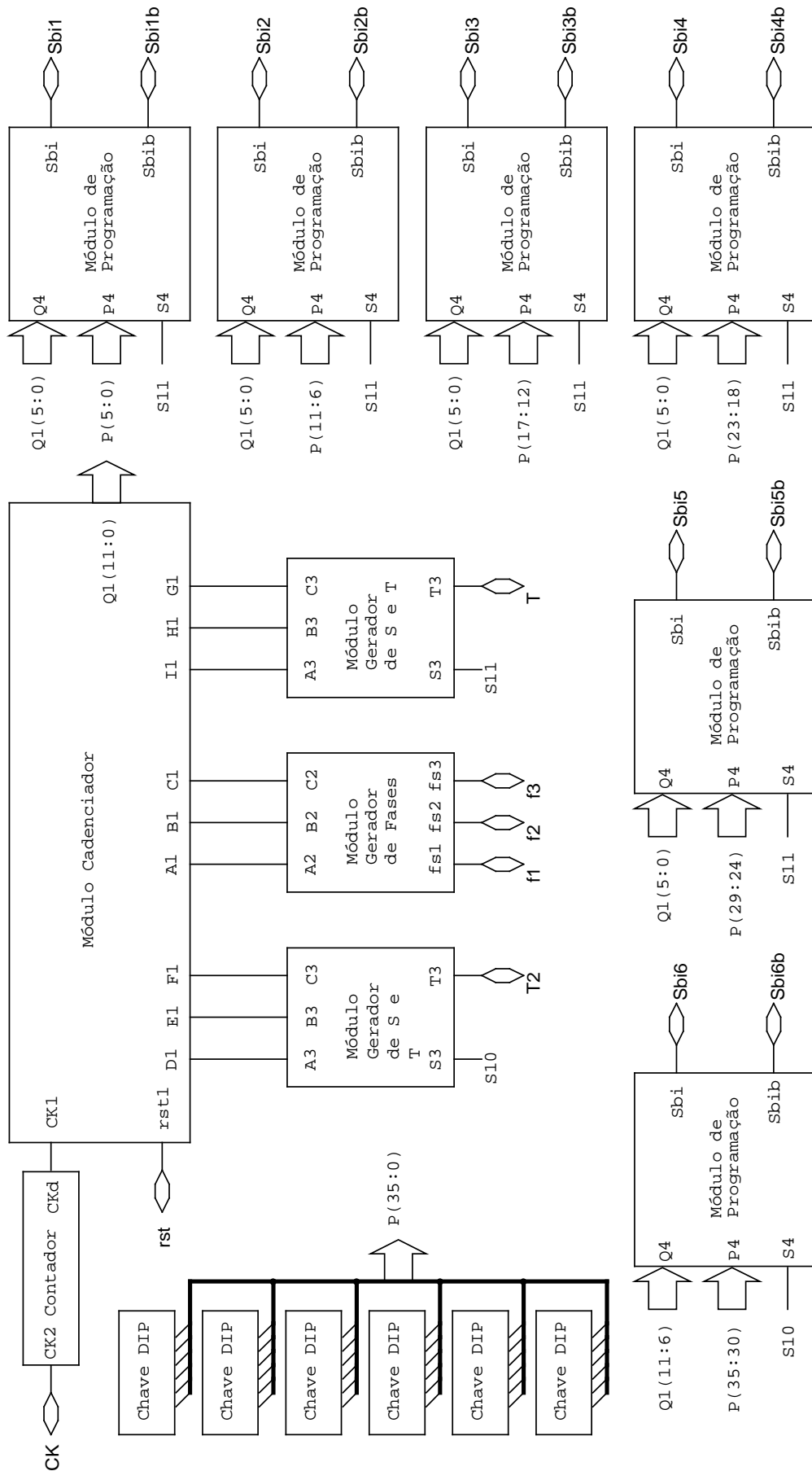


Fig. A.1 : Diagrama em blocos do circuito programado na FPGA.

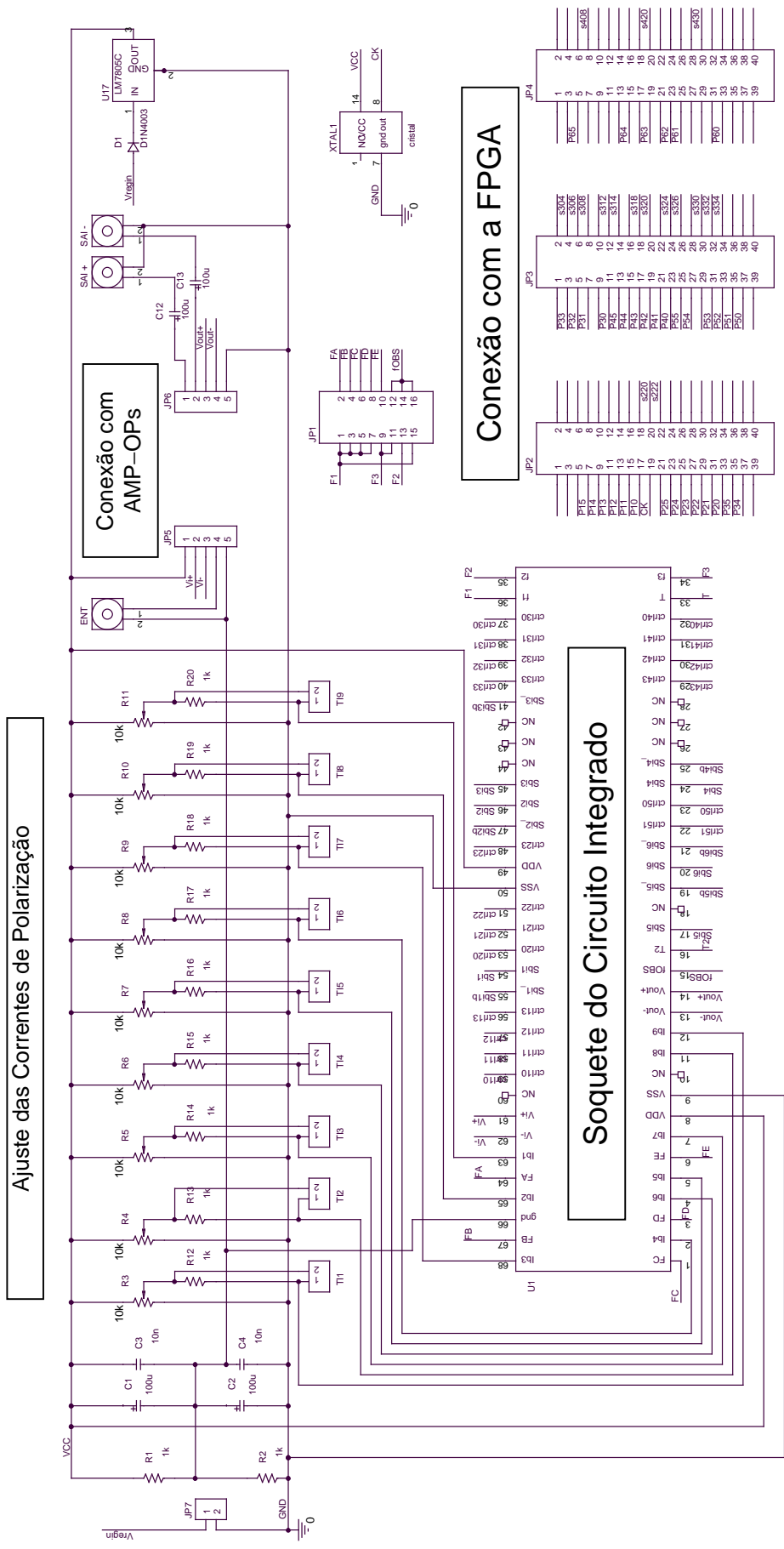


Fig. A.2: Diagrama esquemático da placa de teste (parte I): fontes de corrente, alimentação e conexões.

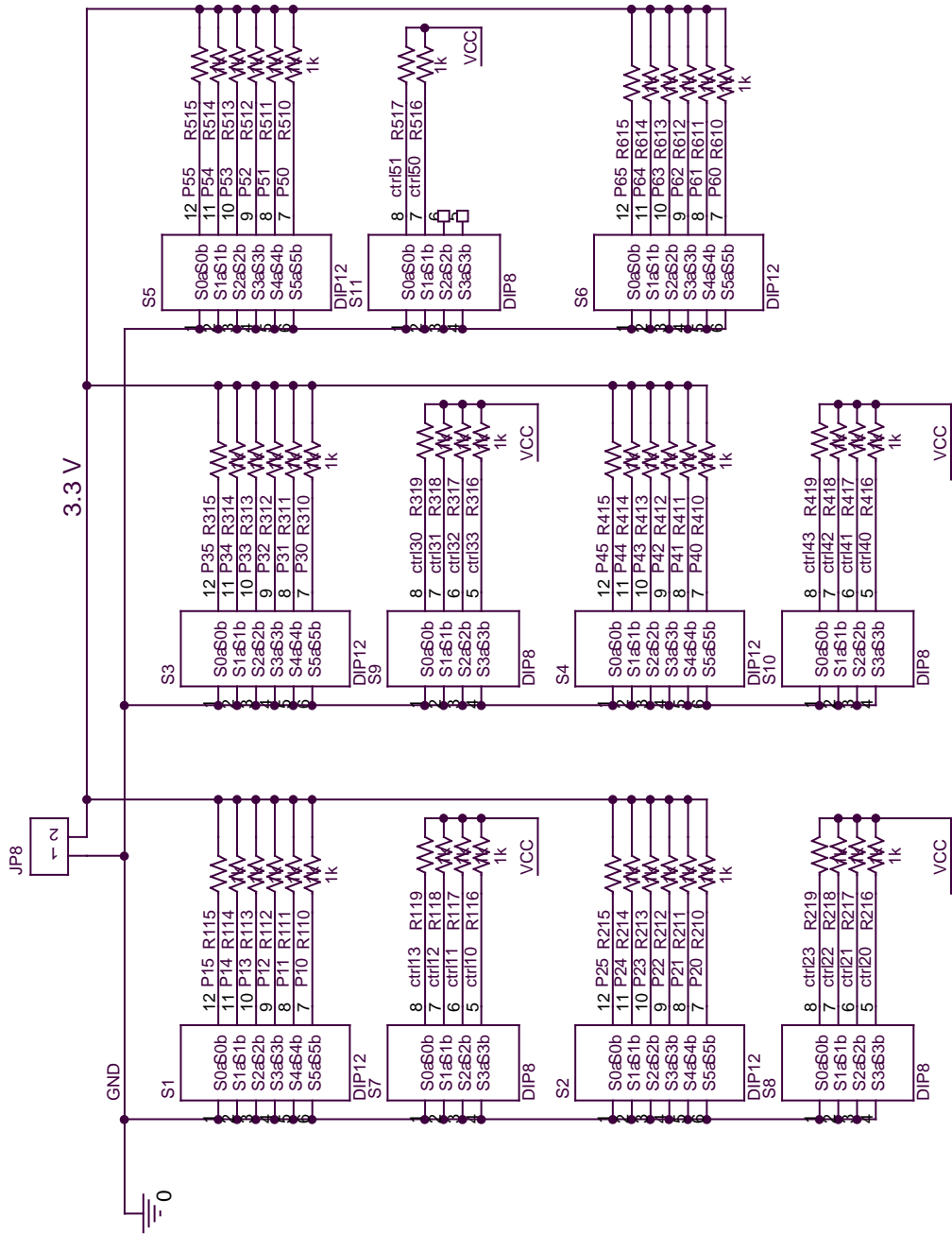


Fig. A.4: Diagrama esquemático da placa de teste (parte III): chaves DIP para programação do filtro.

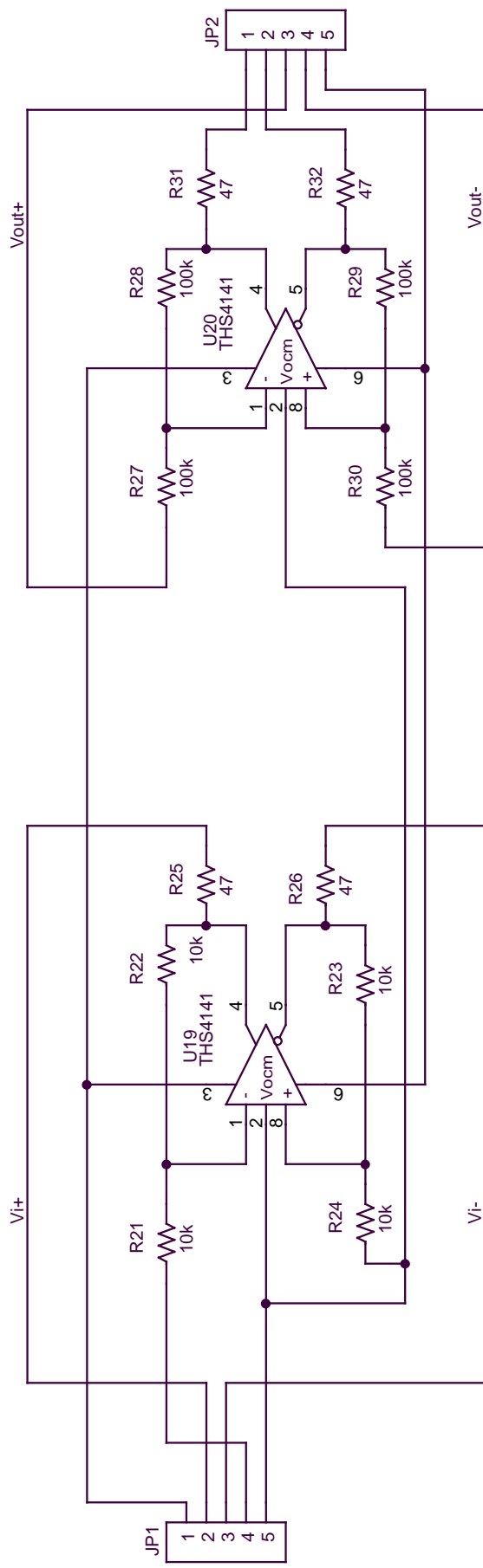


Fig. A.5: Diagrama esquemático da placa de teste (parte IV): amp-ops de entrada e saída.

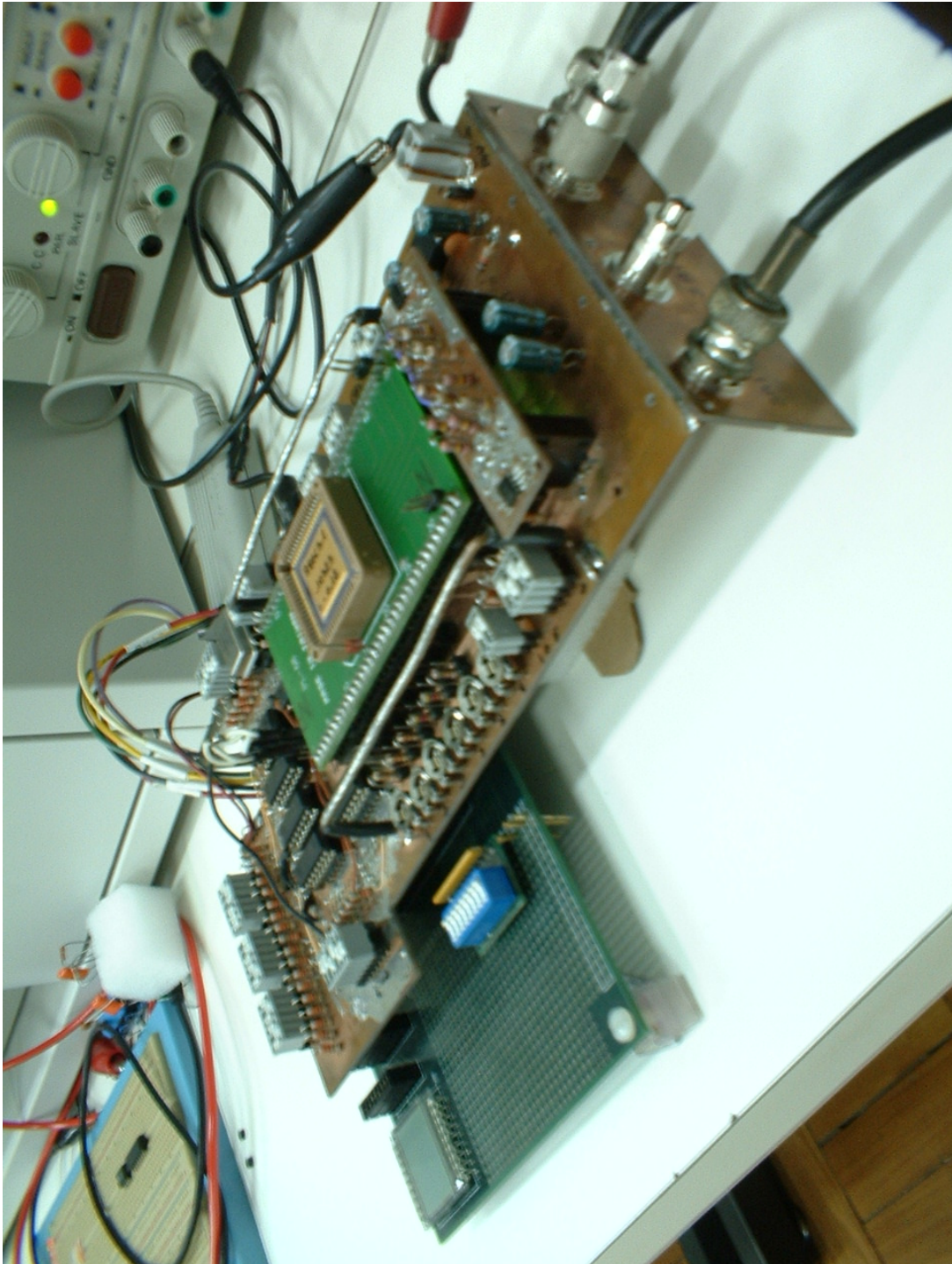


Fig. A.6: Foto da placa de teste

PLACA de TESTE

Este código define as ligações entre os módulos e os sinais de entrada e saída da FPGA, conforme indicado no diagrama em blocos da Fig. A.1.

```
-- Laboratorio de Processamento Analogico e Digital de Sinais
-- Projeto da placa de teste para o ASIC xxxx
-- Entidade superior <PlacTest.vhd> - Faz as ligacoes entre os sinais das entidades inferiores
-- Professor: Joarez Bastos Monteiro
-- Aluno: Rodrigo de Carvalho Mendez
-- Ultima atualizacao: 20/05/2003

library IEEE; -- Usa a biblioteca IEEE
use IEEE.std_logic_1164.all; -- Permite usar os tipos STD_LOGIC

entity PlacTest is -- Define a entidade superior <PlacTest.vhd>, topo da hierarquia
  port(
    CK:    in STD_LOGIC; -- Clock do sistema
    rst:   in STD_LOGIC; -- Reset assincrono do sistema
    P:     in STD_LOGIC_VECTOR (35 downto 0); -- Entradas externas de configuracao dos modulos
    de programacao
    F1:    out STD_LOGIC; -- Saida da fase F1
    F2:    out STD_LOGIC; -- Saida da fase F2
    F3:    out STD_LOGIC; -- Saida da fase F3
    T:     out STD_LOGIC; -- Saida do sinal T
    T2:    out STD_LOGIC; -- Saida do sinal T2
    Sbi1:  out STD_LOGIC; -- Saida do modulo de programacao #1
    Sbi1b: out STD_LOGIC; -- Saida negada do modulo de programacao #1
    Sbi2:  out STD_LOGIC; -- Saida do modulo de programacao #2
    Sbi2b: out STD_LOGIC; -- Saida negada do modulo de programacao #2
    Sbi3:  out STD_LOGIC; -- Saida do modulo de programacao #3
    Sbi3b: out STD_LOGIC; -- Saida negada do modulo de programacao #3
    Sbi4:  out STD_LOGIC; -- Saida do modulo de programacao #4
    Sbi4b: out STD_LOGIC; -- Saida negada do modulo de programacao #4
    Sbi5:  out STD_LOGIC; -- Saida do modulo de programacao #5
    Sbi5b: out STD_LOGIC; -- Saida negada do modulo de programacao #5
    Sbi6:  out STD_LOGIC; -- Saida do modulo de programacao #6
    Sbi6b: out STD_LOGIC; -- Saida negada do modulo de programacao #6
  );
end PlacTest; -- Fim da entidade

architecture PlacTest_Arch of PlacTest is -- Define a arquitetura da entidade
```

-- Componentes de interface com as entidades inferiores na hierarquia

component Contador-- Componente de interface com a entidade <Contador.vhd>

```
port(  
  CK2:  in STD_LOGIC;  
  CKd:  out STD_LOGIC  
);  
end component;
```

component ModCad -- Componente de interface com a entidade <ModCad.vhd>

```
port(  
  CK1:  in STD_LOGIC;  
  rst1: in STD_LOGIC;  
  A1:   out STD_LOGIC;  
  B1:   out STD_LOGIC;  
  C1:   out STD_LOGIC;  
  D1:   out STD_LOGIC;  
  E1:   out STD_LOGIC;  
  F1:   out STD_LOGIC;  
  G1:   out STD_LOGIC;  
  H1:   out STD_LOGIC;  
  I1:   out STD_LOGIC;  
  Q1 :  out STD_LOGIC_VECTOR (11 downto 0)  
);  
end component;
```

component ModGerFa -- Componente de interface com a entidade <ModGerFa.vhd>

```
port(  
  A2:  in STD_LOGIC;  
  B2:  in STD_LOGIC;  
  C2:  in STD_LOGIC;  
  fs1: out STD_LOGIC;  
  fs2: out STD_LOGIC;  
  fs3: out STD_LOGIC  
);  
end component;
```

component ModGerST -- Componente de interface com a entidade <ModGerST.vhd>

```
port(  
  A3:  in STD_LOGIC;  
  B3:  in STD_LOGIC;  
  C3:  in STD_LOGIC;  
  S3:  out STD_LOGIC;  
  T3:  out STD_LOGIC
```

```

    );
end component;

component ModProg -- Componente de interface com a entidade <ModProg.vhd>
port(
    P4:   in STD_LOGIC_VECTOR (5 downto 0);
    Q4:   in STD_LOGIC_VECTOR (5 downto 0);
    S4:   in STD_LOGIC;
    Sbib: out STD_LOGIC;
    Sbi:  out STD_LOGIC
);
end component;

-- Sinais internos da entidade
signal Q1:   STD_LOGIC_VECTOR (11 downto 0); -- Sinais Qmn para os modulos programadores
signal s1, s2, s3: STD_LOGIC; -- Sinais para gerar as fases F1, F2 e F3 em <ModGerFa.vhd>
signal s4, s5, s6: STD_LOGIC; -- Sinais para gerar S2 e T2 em <ModGerST.vhd #2>
signal s7, s8, s9: STD_LOGIC; -- Sinais para gerar S e T em <ModGerST.vhd #1>
signal s10, s11:  STD_LOGIC; -- Mapeamento de S2 e S, respectivamente, indo para os modulos
programadores
signal CKx:   STD_LOGIC; -- Clock apos passar pelo divisor 2^4
signal CKy:   STD_LOGIC; -- Clock antes da divisao

begin

    CKy<=CK;

    MOD0: Contador port map (CK2=>CKy, CKd=>CKx);

    MOD1: ModCad port map (CK1=>CKx, rst1=>rst, Q1 => Q1, A1=>s1, B1=>s2, C1=>s3, D1=>s4,
E1=>s5, F1=>s6, G1=>s7, H1=>s8, I1=>s9);

    MOD2: ModGerFa port map (A2=>s1, B2=>s2, C2=>s3, fs1=>F1, fs2=>F2, fs3=>F3);

    MOD3: ModGerST port map (A3=>s4, B3=>s5, C3=>s6, T3=>T2, S3=>s10);
    MOD4: ModGerST port map (A3=>s9, B3=>s8, C3=>s7, T3=>T, S3=>s11);

    MOD5: ModProg port map (Q4=>Q1( 5 downto 0), P4=>P( 5 downto 0), S4=>s11, Sbi=>Sbi1,
Sbib=>Sbi1b);
    MOD6: ModProg port map (Q4=>Q1( 5 downto 0), P4=>P(11 downto 6), S4=>s11, Sbi=>Sbi2,
Sbib=>Sbi2b);
    MOD7: ModProg port map (Q4=>Q1( 5 downto 0), P4=>P(17 downto 12), S4=>s11, Sbi=>Sbi3,
Sbib=>Sbi3b);
    MOD8: ModProg port map (Q4=>Q1( 5 downto 0), P4=>P(23 downto 18), S4=>s11, Sbi=>Sbi4,
Sbib=>Sbi4b);

```

```

MOD9: ModProg port map (Q4=>Q1( 5 downto 0), P4=>P(29 downto 24), S4=>s11, Sbi=>Sbi5,
Sbib=>Sbi5b);
MOD10: ModProg port map (Q4=>Q1(11 downto 6), P4=>P(35 downto 30), S4=>s10, Sbi=>Sbi6,
Sbib=>Sbi6b);

```

```

end PlacTest_Arch; -- Fim da arquitetura

```

MÓDULO CADENCIADOR

Recebe o sinal de *clock* e gera os sinais sincronizados para todos os demais módulos. Também pode ser reinicializado por acionamento manual (*rst*).

```

-- Laboratorio de Processamento Analogico e Digital de Sinais
-- Projeto da placa de teste para o ASIC xxxx
-- Entidade inferior <ModCad.vhd> - Gera os sinais para todos os modulos posteriores
-- Professor: Joarez Bastos Monteiro
-- Aluno: Rodrigo de Carvalho Mendez
-- Ultima atualizacao: 20/05/2003

library IEEE; -- Usa a biblioteca IEEE
use IEEE.std_logic_1164.all; -- Permite usar os tipos STD_LOGIC

entity ModCad is -- Define a entidade <ModCad.vhd>, secundaria na hierarquia
  port(
    CK1:    in STD_LOGIC; -- Clock do sistema
    rst1:   in STD_LOGIC; -- Reset assincrono do sistema
    A1:     out STD_LOGIC; -- Sinais de saida, sao combinacoes dos sinais do Registrador de
Deslocamento
    B1:     out STD_LOGIC;
    C1:     out STD_LOGIC;
    D1:     out STD_LOGIC;
    E1:     out STD_LOGIC;
    F1:     out STD_LOGIC;
    G1:     out STD_LOGIC;
    H1:     out STD_LOGIC;
    I1:     out STD_LOGIC;
    Q1:     out STD_LOGIC_VECTOR (11 downto 0) -- Sinais Qmn para os modulos programadores
  );
end Modcad; -- Fim da entidade

architecture ModCad_Arch of ModCad is -- Define a arquitetura da entidade

signal REG:    STD_LOGIC_VECTOR(41 downto 0); -- Sinal interno, o Registrador de 42 bits

```

```

begin
-- Processo interno, um Registrador de Deslocamento de 42 bits (ShiftReg)
process (CK1, REG, rst1) -- Caso um destes sinais mude de valor
begin -- Ocorre uma execucao do processo
    if rst1 = '0' then -- Caso haja reset,
        REG <= (40 downto 0 => '1', others => '0'); -- Os 40 Msb do ShiftReg sao colocados em '1' e
os 2 Lsb em '0'
        elsif CK1'event and CK1='1'then -- Senao, numa subida de clock
            REG <= REG(0) & REG(41 downto 1); -- Desloca-se uma vez
        end if;
-- Atualizacao dos valores dos sinais de saida a cada deslocamento do ShiftReg
        Q1 <= (REG(25) & REG(23) & REG(21) & REG(19) & REG(17) & REG(15) & REG(11) & REG(9) &
REG(7) & REG(5) & REG(3) & REG(1));
            A1 <= not (REG(28) and REG(29) and REG(30) and REG(31) and REG(32) and REG(33) and
REG(34) and REG(35) and REG(36) and REG(37) and REG(38) and REG(39) and REG(40) and
REG(41));
            B1 <= not (REG(14) and REG(15) and REG(16) and REG(17) and REG(18) and REG(19) and
REG(20) and REG(21) and REG(22) and REG(23) and REG(24) and REG(25) and REG(26) and
REG(27));
            C1 <= not (REG(0) and REG(1) and REG(2) and REG(3) and REG(4) and REG(5) and REG(6)
and REG(7) and REG(8) and REG(9) and REG(10) and REG(11) and REG(12) and REG(13));
            D1 <= not (REG(14) and REG(15) and REG(16) and REG(17) and REG(18) and REG(19) and
REG(20) and REG(21) and REG(22) and REG(23) and REG(24) and REG(25));
            E1 <= not (REG(15) and REG(17) and REG(19) and REG(21) and REG(23) and REG(25));
            F1 <= not (REG(15) and REG(17) and REG(19) and REG(21) and REG(23) and REG(25));
            G1 <= not (REG(1) and REG(3) and REG(5) and REG(7) and REG(9) and REG(11));
            H1 <= not (REG(1) and REG(3) and REG(5) and REG(7) and REG(9) and REG(11));
            I1 <= not (REG(0) and REG(1) and REG(2) and REG(3) and REG(4) and REG(5) and REG(6)
and REG(7) and REG(8) and REG(9) and REG(10) and REG(11));
        end process; -- Fim do processo interno, ShiftReg

end ModCad_Arch; -- Fim da arquitetura

```

MÓDULO GERADOR de FASES

Recebe os sinais gerados pelo Módulo Cadenciador e gera as fases f_1 , f_2 e f_3 , não entrelaçadas.

```

-- Laboratorio de Processamento Analogico e Digital de Sinais
-- Projeto da placa de teste para o ASIC xxxx
-- Entidade inferior <ModGerFa.vhd> - Gera as Fases F1, F2 e F3
-- Professor: Joarez Bastos Monteiro
-- Aluno: Rodrigo de Carvalho Mendez
-- Ultima atualizacao: 20/05/2003

```



```

library IEEE; -- Usa a biblioteca IEEE
use IEEE.std_logic_1164.all; -- Permite usar os tipos STD_LOGIC

entity ModGerFa is -- Define a entidade <ModGerFa.vhd>, secundaria na hierarquia
    port(
        A2:  in STD_LOGIC; -- Entradas que vem de ModCad
        B2:  in STD_LOGIC;
        C2:  in STD_LOGIC;
        fs1: out STD_LOGIC; -- Saidas das fases F1, F2 e F3, para o ASIC
        fs2: out STD_LOGIC;
        fs3: out STD_LOGIC
    );
end ModGerFa; -- Fim da entidade

architecture ModGerFa_Arch of ModGerFa is -- Define a arquitetura da entidade

    signal f1b, f2b, f3b:  STD_LOGIC; -- Sinais internos, apenas para manipulacao de F1, F2 e F3.

begin
    -- Processo interno, atualiza os valores de saida

    process (A2,B2,C2,f1b,f2b,f3b) -- Caso um destes sinais mude de valor
    begin -- Ocorre uma execucao do processo
        f1b <= (A2 and not(f2b) and not(f3b));
        f2b <= (B2 and not(f1b) and not(f3b));
        f3b <= (C2 and not(f1b) and not(f2b));
        fs1 <= f1b;
        fs2 <= f2b;
        fs3 <= f3b;
    end process; -- Fim do processo interno

end Modgerfa_arch; -- Fim da arquitetura

```

MÓDULO GERADOR de S e T

Recebe os sinais gerados pelo Módulo Cadenciador e gera os sinais S e T ou S2 e T2 não entrelaçados. Os sinais S e S2 são entradas para os Módulos de Programação.

```

-- Laboratorio de Processamento Analogico e Digital de Sinais
-- Projeto da placa de teste para o ASIC xxxx
-- Entidade inferior <ModGerST.vhd> - Gera os sinais S e T ou S2 e T2 (Esta entidade e mapeada duas
vezes em PlacTest.vhd)
-- Professor: Joarez Bastos Monteiro

```

```

-- Aluno: Rodrigo de Carvalho Mendez
-- Ultima atualizacao: 20/05/2003

library IEEE; -- Usa a biblioteca IEEE
use IEEE.std_logic_1164.all; -- Permite usar os tipos STD_LOGIC

entity ModGerST is -- Define a entidade <ModGerST.vhd>, secundaria na hierarquia
  port(
    A3:  in STD_LOGIC; -- Entradas que vem de ModCad
    B3:  in STD_LOGIC;
    C3:  in STD_LOGIC;
    S3:  out STD_LOGIC; -- Saida S ou S2
    T3:  out STD_LOGIC -- Saida T ou T2
  );
end ModGerST; -- Fim da entidade

architecture ModGerST_Arch of ModGerST is -- Define a arquitetura da entidade

  signal Sb, Saux:  STD_LOGIC; -- Sinais internos, apenas para geracao de S e T, ou S2 e T2

begin
  -- Processo interno, atualiza os valores de saida
  process (A3,B3,C3,Sb,Saux) -- Caso um destes sinais mude de valor
  begin -- Ocorre uma execucao do processo
    Saux <= (not C3) nor Sb;
    Sb <= B3 nor Saux;
    T3 <= Sb and A3;
    S3 <= Saux;
  end process; -- Fim do processo interno

end ModGerST_Arch; -- Fim da arquitetura

```

MÓDULO de PROGRAMAÇÃO

Recebe os sinais gerados pelo Módulo Cadenciador e Módulos Geradores de S e T e gera os sinais Sbi e Sbib, não entrelaçados.

```

-- Laboratorio de Processamento Analogico e Digital de Sinais
-- Projeto da placa de teste para o ASIC xxxx
-- Entidade inferior <ModProg.vhd> - Modulo de programacao para o ASIC
-- Professor: Joarez Bastos Monteiro
-- Aluno: Rodrigo de Carvalho Mendez
-- Ultima atualizacao: 20/05/2003

```

```

library IEEE; -- Usa a biblioteca IEEE
use IEEE.std_logic_1164.all; -- Permite usar os tipos STD_LOGIC

entity ModProg is -- Define a entidade <ModProg.vhd>, secundaria na hierarquia
  port(
    P4:   in STD_LOGIC_VECTOR (5 downto 0); -- Entradas vindas de ModCad
    Q4:   in STD_LOGIC_VECTOR (5 downto 0);
    S4:   in STD_LOGIC;
    Sbib: out STD_LOGIC; -- Saida negada do modulo
    Sbi:  out STD_LOGIC -- Saida do modulo
  );
end ModProg; -- Fim da entidade

architecture ModProg_Arch of ModProg is -- Define a arquitetura da entidade

  signal A, B:   STD_LOGIC_VECTOR (5 downto 0); -- Sinais internos, apenas para geracao Sbi e Sbib
  signal C, D:   STD_LOGIC;

begin
  -- Processo interno, atualiza os valores de saida
  process(S4,P4,Q4,A,B,C,D) -- Caso um destes sinais mude de valor
  begin -- Ocorre uma execucao do processo
    A <= P4 or Q4;
    B <= not P4 or Q4;
    C <= not(A(0) and A(1) and A(2) and A(3) and A(4) and A(5));
    D <= not(B(0) and B(1) and B(2) and B(3) and B(4) and B(5));
    Sbib <= S4 and C;
    Sbi <= S4 and D;
  end process; -- Fim do processo interno

end ModProg_Arch; -- Fim da arquitetura

```

MÓDULO CONTADOR

É um divisor de frequência que pode ser programado. Recebe o sinal de *clock* e gera um sinal de frequência desejada para o Módulo Cadenciador.

```

-- Laboratorio de Processamento Analogico e Digital de Sinais
-- Projeto da placa de teste para o ASIC xxxx
-- Entidade inferior <Contador.vhd> - Divisor para a frequencia de clock
-- Professor: Joarez Bastos Monteiro
-- Aluno: Rodrigo de Carvalho Mendez
-- Ultima atualizacao: 11/09/2003

```

```

library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
use IEEE.std_logic_misc.all;
use IEEE.std_logic_unsigned.all;

entity Contador is
  port (
    CK2: in STD_LOGIC;
    CKd: out STD_LOGIC
  );
end Contador;

architecture Contador_arch of Contador is

  signal count : STD_LOGIC_VECTOR(3 downto 0);

begin
  -- <<enter your statements here>>

  process(CK2)
  begin

    if(CK2'event and CK2='1') then
      count<=count + "0001";
    end if;

  end process;

  CKd<=count(2);
end Contador_arch;

```

Apêndice B

Programação de Coeficientes

Tabela B.1: PROGRAMAÇÃO DO COEFICIENTE β (6 bits)

passo	prog	beta
0	000000	0,0000
1	000001	0,0156
2	000010	0,0313
3	000011	0,0469
4	000100	0,0625
5	000101	0,0781
6	000110	0,0938
7	000111	0,1094
8	001000	0,1250
9	001001	0,1406
10	001010	0,1563
11	001011	0,1719
12	001100	0,1875
13	001101	0,2031
14	001110	0,2188
15	001111	0,2344
16	010000	0,2500
17	010001	0,2656
18	010010	0,2813
19	010011	0,2969
20	010100	0,3125
21	010101	0,3281
22	010110	0,3438
23	010111	0,3594
24	011000	0,3750
25	011001	0,3906
26	011010	0,4063
27	011011	0,4219
28	011100	0,4375
29	011101	0,4531
30	011110	0,4688
31	011111	0,4844

passo	prog	beta
32	100000	0,5000
33	100001	0,5156
34	100010	0,5313
35	100011	0,5469
36	100100	0,5625
37	100101	0,5781
38	100110	0,5938
39	100111	0,6094
40	101000	0,6250
41	101001	0,6406
42	101010	0,6563
43	101011	0,6719
44	101100	0,6875
45	101101	0,7031
46	101110	0,7188
47	101111	0,7344
48	110000	0,7500
49	110001	0,7656
50	110010	0,7813
51	110011	0,7969
52	110100	0,8125
53	110101	0,8281
54	110110	0,8438
55	110111	0,8594
56	111000	0,8750
57	111001	0,8906
58	111010	0,9063
59	111011	0,9219
60	111100	0,9375
61	111101	0,9531
62	111110	0,9688
63	111111	0,9844

Tabela B.2: PROGRAMAÇÃO DO COEFICIENTE α (8 bits)

passo	prog	alfa	freq norm	Ganho		passo	prog	alfa	freq norm	Ganho			
				z=1	z=-1					z=1	z=-1		
0	00	000000	-2.0000	0.0000	0.0000	4.0000	64	01	000000	-1.0000	0.3333	1.0000	3.0000
1	00	000001	-1.9844	0.0398	0.0156	3.9844	65	01	000001	-0.9844	0.3362	1.0156	2.9844
2	00	000010	-1.9688	0.0563	0.0313	3.9688	66	01	000010	-0.9688	0.3390	1.0313	2.9688
3	00	000011	-1.9531	0.0691	0.0469	3.9531	67	01	000011	-0.9531	0.3419	1.0469	2.9531
4	00	000100	-1.9375	0.0798	0.0625	3.9375	68	01	000100	-0.9375	0.3447	1.0625	2.9375
5	00	000101	-1.9219	0.0893	0.0781	3.9219	69	01	000101	-0.9219	0.3475	1.0781	2.9219
6	00	000110	-1.9063	0.0978	0.0938	3.9063	70	01	000110	-0.9063	0.3503	1.0938	2.9063
7	00	000111	-1.8906	0.1058	0.1094	3.8906	71	01	000111	-0.8906	0.3531	1.1094	2.8906
8	00	001000	-1.8750	0.1131	0.1250	3.8750	72	01	001000	-0.8750	0.3559	1.1250	2.8750
9	00	001001	-1.8594	0.1201	0.1406	3.8594	73	01	001001	-0.8594	0.3586	1.1406	2.8594
10	00	001010	-1.8438	0.1267	0.1563	3.8438	74	01	001010	-0.8438	0.3614	1.1563	2.8438
11	00	001011	-1.8281	0.1329	0.1719	3.8281	75	01	001011	-0.8281	0.3641	1.1719	2.8281
12	00	001100	-1.8125	0.1389	0.1875	3.8125	76	01	001100	-0.8125	0.3668	1.1875	2.8125
13	00	001101	-1.7969	0.1447	0.2031	3.7969	77	01	001101	-0.7969	0.3696	1.2031	2.7969
14	00	001110	-1.7813	0.1503	0.2188	3.7813	78	01	001110	-0.7813	0.3723	1.2188	2.7813
15	00	001111	-1.7656	0.1556	0.2344	3.7656	79	01	001111	-0.7656	0.3750	1.2344	2.7656
16	00	010000	-1.7500	0.1609	0.2500	3.7500	80	01	010000	-0.7500	0.3776	1.2500	2.7500
17	00	010001	-1.7344	0.1659	0.2656	3.7344	81	01	010001	-0.7344	0.3803	1.2656	2.7344
18	00	010010	-1.7188	0.1709	0.2813	3.7188	82	01	010010	-0.7188	0.3830	1.2813	2.7188
19	00	010011	-1.7031	0.1757	0.2969	3.7031	83	01	010011	-0.7031	0.3857	1.2969	2.7031
20	00	010100	-1.6875	0.1803	0.3125	3.6875	84	01	010100	-0.6875	0.3883	1.3125	2.6875
21	00	010101	-1.6719	0.1849	0.3281	3.6719	85	01	010101	-0.6719	0.3909	1.3281	2.6719
22	00	010110	-1.6563	0.1894	0.3438	3.6563	86	01	010110	-0.6563	0.3936	1.3438	2.6563
23	00	010111	-1.6406	0.1938	0.3594	3.6406	87	01	010111	-0.6406	0.3962	1.3594	2.6406
24	00	011000	-1.6250	0.1981	0.3750	3.6250	88	01	011000	-0.6250	0.3988	1.3750	2.6250
25	00	011001	-1.6094	0.2023	0.3906	3.6094	89	01	011001	-0.6094	0.4014	1.3906	2.6094
26	00	011010	-1.5938	0.2065	0.4063	3.5938	90	01	011010	-0.5938	0.4041	1.4063	2.5938
27	00	011011	-1.5781	0.2106	0.4219	3.5781	91	01	011011	-0.5781	0.4067	1.4219	2.5781
28	00	011100	-1.5625	0.2146	0.4375	3.5625	92	01	011100	-0.5625	0.4093	1.4375	2.5625
29	00	011101	-1.5469	0.2185	0.4531	3.5469	93	01	011101	-0.5469	0.4118	1.4531	2.5469
30	00	011110	-1.5313	0.2224	0.4688	3.5313	94	01	011110	-0.5313	0.4144	1.4688	2.5313
31	00	011111	-1.5156	0.2263	0.4844	3.5156	95	01	011111	-0.5156	0.4170	1.4844	2.5156
32	00	100000	-1.5000	0.2301	0.5000	3.5000	96	01	100000	-0.5000	0.4196	1.5000	2.5000
33	00	100001	-1.4844	0.2338	0.5156	3.4844	97	01	100001	-0.4844	0.4221	1.5156	2.4844
34	00	100010	-1.4688	0.2375	0.5313	3.4688	98	01	100010	-0.4688	0.4247	1.5313	2.4688
35	00	100011	-1.4531	0.2411	0.5469	3.4531	99	01	100011	-0.4531	0.4273	1.5469	2.4531
36	00	100100	-1.4375	0.2447	0.5625	3.4375	100	01	100100	-0.4375	0.4298	1.5625	2.4375
37	00	100101	-1.4219	0.2483	0.5781	3.4219	101	01	100101	-0.4219	0.4323	1.5781	2.4219
38	00	100110	-1.4063	0.2518	0.5938	3.4063	102	01	100110	-0.4063	0.4349	1.5938	2.4063
39	00	100111	-1.3906	0.2553	0.6094	3.3906	103	01	100111	-0.3906	0.4374	1.6094	2.3906
40	00	101000	-1.3750	0.2587	0.6250	3.3750	104	01	101000	-0.3750	0.4400	1.6250	2.3750
41	00	101001	-1.3594	0.2621	0.6406	3.3594	105	01	101001	-0.3594	0.4425	1.6406	2.3594
42	00	101010	-1.3438	0.2655	0.6563	3.3438	106	01	101010	-0.3438	0.4450	1.6563	2.3438
43	00	101011	-1.3281	0.2688	0.6719	3.3281	107	01	101011	-0.3281	0.4475	1.6719	2.3281
44	00	101100	-1.3125	0.2721	0.6875	3.3125	108	01	101100	-0.3125	0.4501	1.6875	2.3125
45	00	101101	-1.2969	0.2754	0.7031	3.2969	109	01	101101	-0.2969	0.4526	1.7031	2.2969
46	00	101110	-1.2813	0.2787	0.7188	3.2813	110	01	101110	-0.2813	0.4551	1.7188	2.2813
47	00	101111	-1.2656	0.2819	0.7344	3.2656	111	01	101111	-0.2656	0.4576	1.7344	2.2656
48	00	110000	-1.2500	0.2851	0.7500	3.2500	112	01	110000	-0.2500	0.4601	1.7500	2.2500
49	00	110001	-1.2344	0.2883	0.7656	3.2344	113	01	110001	-0.2344	0.4626	1.7656	2.2344
50	00	110010	-1.2188	0.2914	0.7813	3.2188	114	01	110010	-0.2188	0.4651	1.7813	2.2188
51	00	110011	-1.2031	0.2945	0.7969	3.2031	115	01	110011	-0.2031	0.4676	1.7969	2.2031
52	00	110100	-1.1875	0.2976	0.8125	3.1875	116	01	110100	-0.1875	0.4701	1.8125	2.1875
53	00	110101	-1.1719	0.3007	0.8281	3.1719	117	01	110101	-0.1719	0.4726	1.8281	2.1719
54	00	110110	-1.1563	0.3038	0.8438	3.1563	118	01	110110	-0.1563	0.4751	1.8438	2.1563
55	00	110111	-1.1406	0.3068	0.8594	3.1406	119	01	110111	-0.1406	0.4776	1.8594	2.1406
56	00	111000	-1.1250	0.3098	0.8750	3.1250	120	01	111000	-0.1250	0.4801	1.8750	2.1250
57	00	111001	-1.1094	0.3128	0.8906	3.1094	121	01	111001	-0.1094	0.4826	1.8906	2.1094
58	00	111010	-1.0938	0.3158	0.9063	3.0938	122	01	111010	-0.0938	0.4851	1.9063	2.0938
59	00	111011	-1.0781	0.3188	0.9219	3.0781	123	01	111011	-0.0781	0.4876	1.9219	2.0781
60	00	111100	-1.0625	0.3217	0.9375	3.0625	124	01	111100	-0.0625	0.4901	1.9375	2.0625
61	00	111101	-1.0469	0.3246	0.9531	3.0469	125	01	111101	-0.0469	0.4925	1.9531	2.0469
62	00	111110	-1.0313	0.3276	0.9688	3.0313	126	01	111110	-0.0313	0.4950	1.9688	2.0313
63	00	111111	-1.0156	0.3305	0.9844	3.0156	127	01	111111	-0.0156	0.4975	1.9844	2.0156

PROGRAMAÇÃO DO COEFICIENTE α (8 bits) (continuação)

passo	prog	alfa	freq norm	Ganho		passo	prog	alfa	freq norm	Ganho			
				z=1	z=-1					z=1	z=-1		
128	10	000000	0,0000	0,5000	2,0000	2,0000	192	11	000000	1,0000	0,6667	3,0000	1,0000
129	10	000001	0,0156	0,5025	2,0156	1,9844	193	11	000001	1,0156	0,6695	3,0156	0,9844
130	10	000010	0,0313	0,5050	2,0313	1,9688	194	11	000010	1,0313	0,6724	3,0313	0,9688
131	10	000011	0,0469	0,5075	2,0469	1,9531	195	11	000011	1,0469	0,6754	3,0469	0,9531
132	10	000100	0,0625	0,5099	2,0625	1,9375	196	11	000100	1,0625	0,6783	3,0625	0,9375
133	10	000101	0,0781	0,5124	2,0781	1,9219	197	11	000101	1,0781	0,6812	3,0781	0,9219
134	10	000110	0,0938	0,5149	2,0938	1,9063	198	11	000110	1,0938	0,6842	3,0938	0,9063
135	10	000111	0,1094	0,5174	2,1094	1,8906	199	11	000111	1,1094	0,6872	3,1094	0,8906
136	10	001000	0,1250	0,5199	2,1250	1,8750	200	11	001000	1,1250	0,6902	3,1250	0,8750
137	10	001001	0,1406	0,5224	2,1406	1,8594	201	11	001001	1,1406	0,6932	3,1406	0,8594
138	10	001010	0,1563	0,5249	2,1563	1,8438	202	11	001010	1,1563	0,6962	3,1563	0,8438
139	10	001011	0,1719	0,5274	2,1719	1,8281	203	11	001011	1,1719	0,6993	3,1719	0,8281
140	10	001100	0,1875	0,5299	2,1875	1,8125	204	11	001100	1,1875	0,7024	3,1875	0,8125
141	10	001101	0,2031	0,5324	2,2031	1,7969	205	11	001101	1,2031	0,7055	3,2031	0,7969
142	10	001110	0,2188	0,5349	2,2188	1,7813	206	11	001110	1,2188	0,7086	3,2188	0,7813
143	10	001111	0,2344	0,5374	2,2344	1,7656	207	11	001111	1,2344	0,7117	3,2344	0,7656
144	10	010000	0,2500	0,5399	2,2500	1,7500	208	11	010000	1,2500	0,7149	3,2500	0,7500
145	10	010001	0,2656	0,5424	2,2656	1,7344	209	11	010001	1,2656	0,7181	3,2656	0,7344
146	10	010010	0,2813	0,5449	2,2813	1,7188	210	11	010010	1,2813	0,7213	3,2813	0,7188
147	10	010011	0,2969	0,5474	2,2969	1,7031	211	11	010011	1,2969	0,7246	3,2969	0,7031
148	10	010100	0,3125	0,5499	2,3125	1,6875	212	11	010100	1,3125	0,7279	3,3125	0,6875
149	10	010101	0,3281	0,5525	2,3281	1,6719	213	11	010101	1,3281	0,7312	3,3281	0,6719
150	10	010110	0,3438	0,5550	2,3438	1,6563	214	11	010110	1,3438	0,7345	3,3438	0,6563
151	10	010111	0,3594	0,5575	2,3594	1,6406	215	11	010111	1,3594	0,7379	3,3594	0,6406
152	10	011000	0,3750	0,5600	2,3750	1,6250	216	11	011000	1,3750	0,7413	3,3750	0,6250
153	10	011001	0,3906	0,5626	2,3906	1,6094	217	11	011001	1,3906	0,7447	3,3906	0,6094
154	10	011010	0,4063	0,5651	2,4063	1,5938	218	11	011010	1,4063	0,7482	3,4063	0,5938
155	10	011011	0,4219	0,5677	2,4219	1,5781	219	11	011011	1,4219	0,7517	3,4219	0,5781
156	10	011100	0,4375	0,5702	2,4375	1,5625	220	11	011100	1,4375	0,7553	3,4375	0,5625
157	10	011101	0,4531	0,5727	2,4531	1,5469	221	11	011101	1,4531	0,7589	3,4531	0,5469
158	10	011110	0,4688	0,5753	2,4688	1,5313	222	11	011110	1,4688	0,7625	3,4688	0,5313
159	10	011111	0,4844	0,5779	2,4844	1,5156	223	11	011111	1,4844	0,7662	3,4844	0,5156
160	10	100000	0,5000	0,5804	2,5000	1,5000	224	11	100000	1,5000	0,7699	3,5000	0,5000
161	10	100001	0,5156	0,5830	2,5156	1,4844	225	11	100001	1,5156	0,7737	3,5156	0,4844
162	10	100010	0,5313	0,5856	2,5313	1,4688	226	11	100010	1,5313	0,7776	3,5313	0,4688
163	10	100011	0,5469	0,5882	2,5469	1,4531	227	11	100011	1,5469	0,7815	3,5469	0,4531
164	10	100100	0,5625	0,5907	2,5625	1,4375	228	11	100100	1,5625	0,7854	3,5625	0,4375
165	10	100101	0,5781	0,5933	2,5781	1,4219	229	11	100101	1,5781	0,7894	3,5781	0,4219
166	10	100110	0,5938	0,5959	2,5938	1,4063	230	11	100110	1,5938	0,7935	3,5938	0,4063
167	10	100111	0,6094	0,5986	2,6094	1,3906	231	11	100111	1,6094	0,7977	3,6094	0,3906
168	10	101000	0,6250	0,6012	2,6250	1,3750	232	11	101000	1,6250	0,8019	3,6250	0,3750
169	10	101001	0,6406	0,6038	2,6406	1,3594	233	11	101001	1,6406	0,8062	3,6406	0,3594
170	10	101010	0,6563	0,6064	2,6563	1,3438	234	11	101010	1,6563	0,8106	3,6563	0,3438
171	10	101011	0,6719	0,6091	2,6719	1,3281	235	11	101011	1,6719	0,8151	3,6719	0,3281
172	10	101100	0,6875	0,6117	2,6875	1,3125	236	11	101100	1,6875	0,8197	3,6875	0,3125
173	10	101101	0,7031	0,6143	2,7031	1,2969	237	11	101101	1,7031	0,8243	3,7031	0,2969
174	10	101110	0,7188	0,6170	2,7188	1,2813	238	11	101110	1,7188	0,8291	3,7188	0,2813
175	10	101111	0,7344	0,6197	2,7344	1,2656	239	11	101111	1,7344	0,8341	3,7344	0,2656
176	10	110000	0,7500	0,6224	2,7500	1,2500	240	11	110000	1,7500	0,8391	3,7500	0,2500
177	10	110001	0,7656	0,6250	2,7656	1,2344	241	11	110001	1,7656	0,8444	3,7656	0,2344
178	10	110010	0,7813	0,6277	2,7813	1,2188	242	11	110010	1,7813	0,8497	3,7813	0,2188
179	10	110011	0,7969	0,6304	2,7969	1,2031	243	11	110011	1,7969	0,8553	3,7969	0,2031
180	10	110100	0,8125	0,6332	2,8125	1,1875	244	11	110100	1,8125	0,8611	3,8125	0,1875
181	10	110101	0,8281	0,6359	2,8281	1,1719	245	11	110101	1,8281	0,8671	3,8281	0,1719
182	10	110110	0,8438	0,6386	2,8438	1,1563	246	11	110110	1,8438	0,8733	3,8438	0,1563
183	10	110111	0,8594	0,6414	2,8594	1,1406	247	11	110111	1,8594	0,8799	3,8594	0,1406
184	10	111000	0,8750	0,6441	2,8750	1,1250	248	11	111000	1,8750	0,8869	3,8750	0,1250
185	10	111001	0,8906	0,6469	2,8906	1,1094	249	11	111001	1,8906	0,8942	3,8906	0,1094
186	10	111010	0,9063	0,6497	2,9063	1,0938	250	11	111010	1,9063	0,9022	3,9063	0,0938
187	10	111011	0,9219	0,6525	2,9219	1,0781	251	11	111011	1,9219	0,9107	3,9219	0,0781
188	10	111100	0,9375	0,6553	2,9375	1,0625	252	11	111100	1,9375	0,9202	3,9375	0,0625
189	10	111101	0,9531	0,6581	2,9531	1,0469	253	11	111101	1,9531	0,9309	3,9531	0,0469
190	10	111110	0,9688	0,6610	2,9688	1,0313	254	11	111110	1,9688	0,9437	3,9688	0,0313
191	10	111111	0,9844	0,6638	2,9844	1,0156	255	11	111111	1,9844	0,9602	3,9844	0,0156

PROGRAMA FONTE: Calc_coef_filtro.m

```
%      F1      F2      F3      F4
%rdB  1.0      3.0      1.0      3.0
%pb   0.30 0.25  0.15  0.30
%rb   0.44 0.27  0.20  0.35
%aten 51      23      25      35

% calculo do filtro usando iirremez
rdB=3;
pb=0.3;
rb=0.35;
[q,r,f,dB]=iirremez(8,2,rdB,pb,rb);

%calculo das raizes do polinomio do numerador
rz=roots(q);

%ZEROS
%formando os polinomios de 2a. ordem
aux=[poly([rz(1) rz(2)]);
     poly([rz(3) rz(4)]);
     poly([rz(5) rz(6)]);
     poly([rz(7) rz(8)])];

% ordenando os zeros
[xx1,yy1]=find(aux(:,2)>=0);
aux1=aux(xx1,:);
[xxx1,ind1]=sort(aux1(:,2));
aux1=aux1(ind1,:);
[xx2,yy2]=find(aux(:,2)<0);
aux2=aux(xx2,:);
[xxx2,ind2]=sort(aux2(:,2));
aux2=aux2(flipud(ind2),:);
aux=[aux1;aux2];
aux(:,[1 3])=ones(4,2);
aux=[aux sum(aux')];

%separando alfa_z e ganho em z=1
alfa_z=aux(:,2);
ganho_z=aux(:,4);

%consultando a tabela de coeficientes para definir a programacao do filtro
aux2=-2:1/64:2;
[xx,ind1]=min(abs(aux2-alfa_z(1)));
```



```

bin_alfa_p=dec2bin(ind5-1, 8);
ganho_p=sum(r);

%Mostrando o resultado dos polos
disp('POLOS: coeficientes sem aproximacao')
disp(num2str(r))
disp('alfa_p beta ganho_p')
disp([num2str(alfa_p) [' ' ]...
      num2str(beta) [' ' ]...
      num2str(sum(r))])

%calculando frequencia normalizada do polo
freq_norm_p=atan2(sqrt(4-aux2(ind5).^2)/2,-aux2(ind5)/2)/pi;

%calculando o alfa_p*beta equivalente
coef_eq=aux4(ind6)*aux2(ind5);
ganho=1+coef_eq+aux4(ind6);

disp('POLOS: coeficientes com aproximacao')
disp('      ctrl ')
disp('  ind_tab 10  bin  val_tab  freq_norm  ganho_p ')
disp([ char(['alfa_p '])...
      num2str(ind5-1) [' ' ]...
      char(bin_alfa_p(1:2)) [' ' ]...
      char(bin_alfa_p(3:8)) [' ' ]...
      num2str(aux2(ind5)) [' ' ]...
      num2str(freq_norm_p) [' -']...
      ])
disp([ char(['beta '])...
      num2str(ind6-1) [' ' ]...
      char('xx') [' ' ]...
      char(bin_beta) [' ' ]...
      num2str(aux4(ind6)) [' ' ]...
      num2str('- -') ])
disp([ char(['coef_eq '])...
      ['- - - ' ]...
      num2str(coef_eq) [' ' ]...
      ['- ' ] num2str(ganho) [' ' ]])

```

FILTRO 1

Característica do filtro:

ripple na banda passante

rdB=1;

frequência de corte

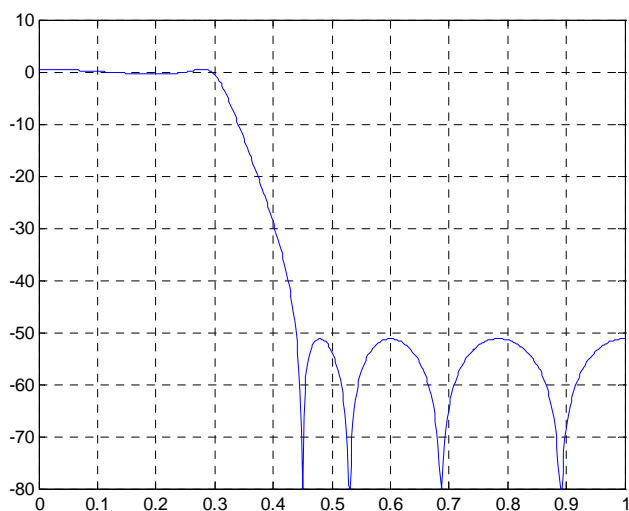
pb=0,3;

início da banda de rejeição

rb=0.44;

Atenuação

51.3 dB



ZEROS:

coeficientes sem aproximação

alfa_z	ganho_z
0.18767	2.1877
1.1063	3.1063
1.8832	3.8832
-0.31469	1.6853

coeficientes com aproximação

ind_tab	aten_z	ctrl		prog	alfa_z	freq_norm
		32	10	zeros		
140	3	10	10	001100	0.1875	0.52989
199	4	11	11	000111	1.1094	0.68716
249	4	11	11	111001	1.8906	0.89424
108	2	01	01	101100	-0.3125	0.45006

PÓLOS:

coeficientes sem aproximação

1 -1.0524 0.8084

alfa_p	beta_p	ganho_p
-1.2953	0.8084	0.75598

coeficientes com aproximação

	ind_tab	ctrl	bin	val_tab	freq_norm	ganho_p
alfa_p	45	00	101101	-1.2969	0.27542	-
beta	52	xx	110100	0.8125	-	-
coef_eq	-	-	-	-1.0537	-	0.75879

FILTRO 2

Característica do filtro:

ripple na banda passante

rdB=3;

freqüência de corte

pb=0,25;

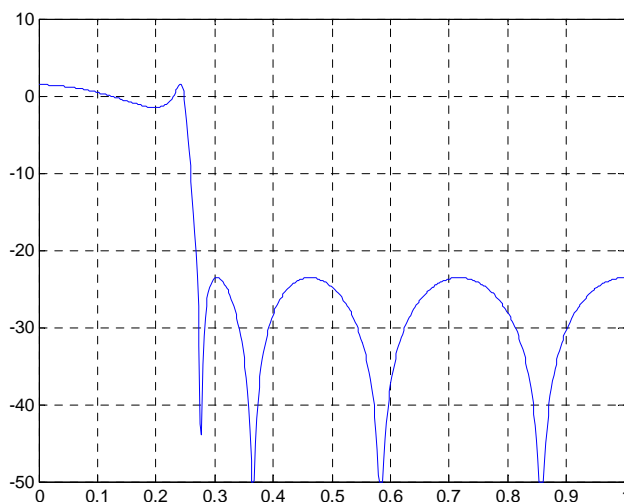
início da banda de rejeição

rb=0.27;

Atenuação

23.6 dB

ZEROS:



coeficientes sem aproximação

alfa_z	ganho_z
0.51891	2.5189
1.8022	3.8022
-0.82076	1.1792
-1.2896	0.71042

coeficientes com aproximação

ind_tab	aten_z	ctrl		prog	alfa_z	freq_norm
		32	10	zeros		
161	3	10	10	100001	0.51563	0.5830
243	4	11	11	110011	1.7969	0.8553
75	2	01	01	001011	-0.82813	0.36411
45	1	00	00	101101	-1.2969	0.27542

PÓLOS:

coeficientes sem aproximação

1 -1.3894 0.94178

alfa_p	beta_p	ganho_p
-1.4821	0.94178	0.55233

coeficientes com aproximação

	ind_tab	ctrl	bin	val_tab	freq_norm	ganho_p
alfa_p	33	00	100001	-1.4844	0.23379	-
beta	60	xx	111100	0.9375	-	-
coef_eq	-	-	-	-1.3916	-	0.5459

FILTRO 3

Característica do filtro:

ripple na banda passante

rdB=1;

freqüência de corte

pb=0,15;

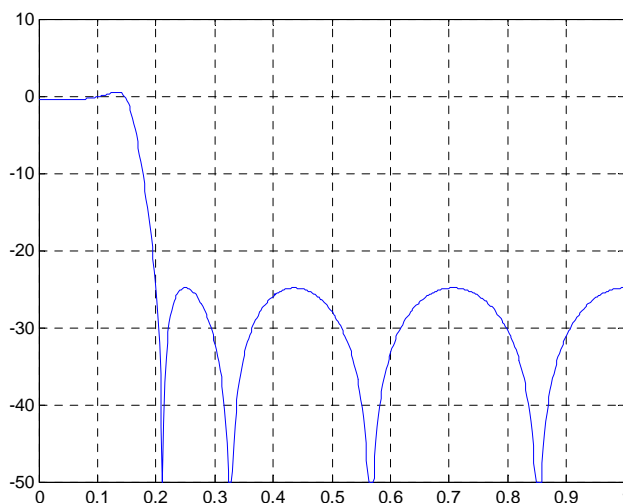
início da banda de rejeição

rb=0.20;

Atenuação

24.9 dB

ZEROS:



coeficientes sem aproximação

alfa_z	ganho_z
0.41373	2.4137
1.7885	3.7885
-1.0353	0.96466
-1.5757	0.42428

coeficientes com aproximação

ind_tab	aten_z	ctrl		prog	alfa_z	freq_norm
		32	10	zeros		
154	3	10	10	011010	0.40625	0.56511
242	4	11	11	110010	1.7813	0.84973
62	1	00	00	111110	-1.0313	0.32756
27	1	00	00	011011	-1.5781	0.21057

PÓLOS:

coeficientes sem aproximação

1 -1.6338 0.84978

alfa_p	beta_p	ganho_p
-1.9363	0.84978	0.21599

coeficientes com aproximação

	ind_tab	ctrl	bin	val_tab	freq_norm	ganho_p
alfa_p	4	00	000100	-1.9375	0.079786	-
beta	54	xx	110110	0.84375	-	-
coef_eq	-	-	-	-1.6348	-	0.20898

FILTRO 4

Característica do filtro:

ripple na banda passante

rdB=3;

freqüência de corte

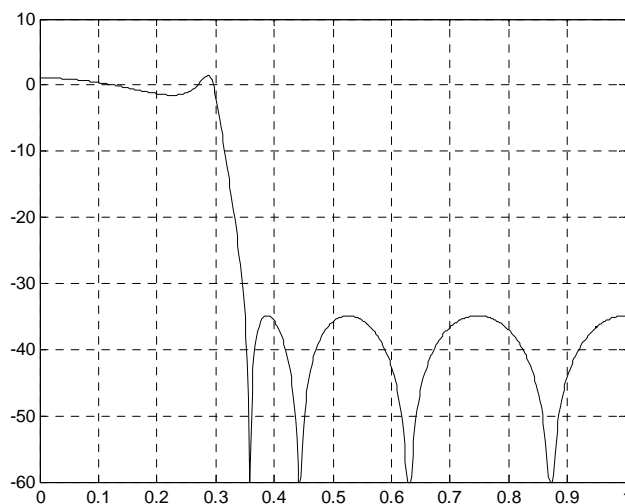
pb=0,3;

início da banda de rejeição

rb=0.35;

Atenuação

34.9 dB



ZEROS:

coeficientes sem aproximação

alfa_z	ganho_z
0.79292	2.7929
1.8398	3.8398
-0.35299	1.647
-0.85876	1.1412

coeficientes com aproximação

ind_tab	aten_z	ctrl		prog	alfa_z	freq_norm
		32	10	zeros		
179	3	10	10	110011	0.79688	0.63045
246	4	11	11	110110	1.8438	0.87334
105	2	01	01	101001	-0.35938	0.44249
73	2	01	01	001001	-0.85938	0.35862

PÓLOS:

coeficientes sem aproximação

1 -1.1548 0.91138

alfa_p	beta_p	ganho_p
-1.2743	0.91138	0.75656

coeficientes com aproximação

	ind_tab	ctrl	bin	val_tab	freq_norm	ganho_p
alfa_p	46	00	101110	-1.2813	0.27868	-
beta	58	xx	111010	0.90625	-	-
coef_eq	-	-	-	-1.1611	-	0.74512

Referências

- [1] GREGORIAN, R., TEMES, G.C., Analog MOS Integrated Circuits for Signal Processing, 1 ed., Wiley, New York – USA, 1986.
- [2] GREGORIAN, R., MARTIN, K.W., TEMES, G.C., “Switched-Capacitor Circuit Design”, *Proceedings of the IEEE*, vol. 71, No. 8, pp. 941-966, 1983.
- [3] PAULINO, N., FRANCA, J. E., MARTINS, F. P., “Programmable CMOS Switched-Capacitor Biquad Using Quasi-Passive Algorithmic DAC's”, *IEEE Journal of Solid-State Circuits*, Vol. 30, No. 6, pp. 715-719, 1995.
- [4] MOON, U., “CMOS High-Frequency Switched-Capacitor Filters for Telecommunication Applications”, *IEEE Journal of Solid-State Circuits*, Vol. 35, No. 2, pp. 212-220, 2000.
- [5] MARTINS, F. P., PAULINO, N. F., FRANCA, J. E., “Charge Programming Techniques for SC Biquads”, *Proc. IEEE ISCAS93*, Chicago, IL, pp. 1160-1163, 1993.
- [6] SILVA-MARTINEZ, J., “A Programmable Switched-Capacitor Filter” *IEEE Int. Symposium on Circuits and Systems*, ISCAS '94., Vol. 5, pp. 727 – 730, 1994.
- [7] DUPPILS, M.; EKLUND, J.-E.; SVENSSON, C., “A Novel Mixed Analog/Digital MAC Unit Implemented With SC Technique Suitable for Fully Programmable Narrow-Band FIR Filter Applications”, *Proc. IEEE ICECS'99*, Vol. 3, pp. 1197-1200, 1999.
- [8] DUQUE-CARRILLO, J. F., SILVA-MARTINEZ, J., SÁNCHEZ-SINENCIO, E., “Programmable Switched-Capacitor Bump Equalizer Architecture”, *IEEE Journal of Solid-State Circuits*, Vol. 25, No. 4, pp. 1035-1039, 1990.

- [9] PETRAGLIA, A., MITRA, S. K., "Switched-Capacitor Equalizer with Digitally Programmable Tuning Characteristics", *IEEE Transactions on Circuits and Systems*, vol. 38, No. 11, pp. 1322-1331, 1991.
- [10] PETRAGLIA, A. E PEREIRA, J. S., " Switched-Capacitor Decimation Filters with Direct-Form Polyphase Structure Having Very Small Sensitivity Characteristics", *Proc. Int. Symp. on Circuits and Systems*, Orlando, FL, EUA, pp. II.73-II.76, 1999.
- [11] PETRAGLIA, A., "Fundamental frequency response bounds of direct form switched-capacitor filters with capacitance mismatch", *IEEE Trans. on Circuits and Systems – Part II: Analog and Digital Signal Processing*, Vol. 48, pp. 340-350, Apr. 2001.
- [12] PEREIRA, J.S.; PETRAGLIA, A., "Low-Sensitivity Direct-Form IIR SC Filters With Improved Phase Linearity", *IEEE Int. Symp. on Circuits and Systems*, Vol. 3, pp. 169 – 172, 2000.
- [13] AUSIN, J.L., DUQUE-CARRILLO, J. F., TORRELLI, G., SÁNCHEZ-SINENCIO, E., "Switched-Capacitor Circuits with Periodical Non-Uniform Individual Sampling", *IEEE Transactions on Circuits and System – II: Analog and Digital Signal Processing*, vol. 50, No. 8 pp. 404-414, 2003.
- [14] MONTEIRO, J. B., PETRAGLIA, A., LEME, C. A., "A Digitally Programmable IIR Switched-Capacitor Filter for CMOS Technology", *IEEE Int. Symp. on Circuits and Systems*, Sidney, Australia, pp. I.69-I.72, May 2001.
- [15] MONTEIRO, J. B., PETRAGLIA, A., "A 0.8 μ m CMOS Programmable IIR SC Filter", *IEEE Int. Symp. on Circuits and Systems*, Toronto, Canadá, May 2004 - Aceito para apresentação.
- [16] FRIED, D. L., "Analog Sample-Data Filters", *IEEE Journal of Solid-State Circuits*, Vol. 7, No. 4, pp. 302-304, 1972.
- [17] FISCHER, G., "Analog FIR Filters by Switched-Capacitor Techniques", *IEEE Transactions on Circuits and Systems*, vol. 37, No. 6, pp. 808-814, 1990.

- [18] FISCHER, G., "Switched-Capacitor FIR Filters – A Feasibility Study", *IEEE Transactions on Circuits and System – II: Analog and Digital Signal Processing*, vol.41, No. 12, pp. 823-827, 1994.
- [19] HASLER, M., "Stray Insensitive Switched Capacitor Filters", *IEEE Int. Symp. on Circuits and Systems*, pp. 195-197, 1981.
- [20] BARUQUI, F. P., "Estruturas a Capacitores Chaveados de Baixa Sensibilidade para o Processamento de Sinais", Tese de doutorado COPPE/UFRJ, 1999.
- [21] REN, Y., LEUNG, B. H., LIN Y., "A Mismatch-Independent DNL Pipelined Analog-to-Digital Converter", *IEEE Transactions on Circuits and System – II: Analog and Digital Signal Processing*, vol. 46, No. 5 pp. 517-526, 1999.
- [22] GUPTA, S. C., HASDORFF, L., *Fundamentals of Automatic Control*, 1 ed., Wiley, New York – USA, 1970.
- [23] JACKSON, L. B., "An Improved Martinez/Parks Algorithm for IIR Design with Unequal Numbers of Poles and Zeros", *IEEE Transactions on Signal Processing*, vol. 42, No. 5, pp. 1234-1238, 1994.