

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
ESCOLA DE ENGENHARIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**TIAGO ROBERTO BALEN**

**TESTE DE DISPOSITIVOS ANALÓGICOS  
PROGRAMÁVEIS (FPAAS)**

Porto Alegre

2006

**TIAGO ROBERTO BALEN**

**TESTE DE DISPOSITIVOS ANALÓGICOS PROGRAMÁVEIS  
(FPAAS)**

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para a obtenção do título de Mestre em Engenharia Elétrica.

Área de concentração: Automação e Instrumentação Eletro-Eletrônica

**ORIENTADOR: Marcelo Soares Lubaszewski**

Porto Alegre

2006

TIAGO ROBERTO BALEN

## **TESTE DE DISPOSITIVOS ANALÓGICOS PROGRAMÁVEIS (FPAAS)**

Esta dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia Elétrica e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Orientador: \_\_\_\_\_

Prof. Dr. Marcelo Soares Lubaszewski, UFRGS

Doutor pela Institut National Polytechnique de Grenoble, França

Banca Examinadora:

Dr. José Vicente Calvano, IPqM

Doutor pela Universidade Federal do Rio de Janeiro – Rio de Janeiro, Brasil

Prof. Dr. Eric Ericson Fabris, UFRGS

Doutor pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Prof. Dra. Fernanda Gusmão de Lima Kastesmidt, UFRGS

Doutora pela Universidade Federal do Rio Grande do Sul – Porto Alegre, Brasil

Coordenador do PPGEE: \_\_\_\_\_

Prof. Dr. Marcelo Soares Lubaszewski

Porto Alegre, Abril de 2006

## **DEDICATÓRIA**

Dedico este trabalho aos meus pais, Altair e Elisete, e à minha amada Maríndia, pois tenho certeza que, mais do que eu, eles estão felizes e orgulhosos desta conquista.

## **AGRADECIMENTOS**

Agradeço a Universidade Federal do Rio Grande do Sul e ao Programa de Pós Graduação em Engenharia Elétrica pelo ensino de qualidade. Agradeço a todos os servidores do Departamento de Engenharia Elétrica em especial aos secretários Manoel Eduardo Hoffmeister e Miriam Rosek e ao encarregado da manutenção, José Roberto Farias, pelo auxílio nas questões burocráticas e pela manutenção do nosso local de trabalho. Agradeço a todos os professores com os quais tive aulas durante o período do Mestrado, em especial aos professores Marcelo Lubaszewski, Luigi Carro, Altamiro Susin e Érika Cota. Não poderia deixar de agradecer meus colegas de laboratório (LaProT) e de disciplinas, Antônio de Quadros Andrade Jr., Érik Schüller, Gustavo Vieira Pereira, Josias Otaciel Mainairdi e Carla Diniz Lopes, pelo companheirismo e pronta disposição a ajudar quando necessário. Agradeço também ao Marcelo Negreiros pela ajuda com a rede e com os computadores e a todo pessoal do LaPsi, pelo empréstimo de equipamentos. Também agradeço aos ex-bolsistas Michael Schreiber e Tiago Jost pelas suas contribuições a este trabalho.

Agradeço a CAPES pela provisão da bolsa de Mestrado e ao CNPq, PPGEE, PROPESQ e FAPERGS pelos auxílios financeiros para participação em eventos científicos.

Agradeço de forma especial meu orientador, Marcelo Lubaszewski, pelo constante incentivo ao meu trabalho, pela confiança e pelo exemplo profissional e pessoal. Agradeço também aos amigos de Erechim pelos momentos alegres e festivos e por torcerem pelo meu sucesso.

Agradeço a toda minha família em especial meus pais Altair Balen e Elisete Balen e meu irmão, Fabiano Balen pelo suporte afetivo, pelo carinho, pelo apoio financeiro durante a graduação, pelo orgulho de ter um acadêmico na família, e por tudo que eles representam para mim.

Agradeço de forma muito especial à minha noiva, Maríndia Deprá, pelo carinho, compreensão, companheirismo, incentivo e cumplicidade nos bons e maus momentos desta jornada e de outras que passaram (e das muitas que ainda virão).

Agradeço a todos que direta ou indiretamente tornaram possível a realização deste trabalho.

## RESUMO

Neste trabalho o teste de dispositivos analógicos programáveis é abordado. Diversas metodologias de teste analógico existentes são estudadas e algumas delas são utilizadas nas estratégias desenvolvidas. Dois FPAA (*Field Programmable Analog Arrays*) comerciais de fabricantes e modelos distintos são utilizados para validar as estratégias de teste propostas. O primeiro dispositivo estudado é um FPAA de tempo contínuo (capaz de implementar circuitos contínuos no tempo) da *Lattice Semiconductors*. Tal dispositivo é marcado pela característica estrutural de sua programabilidade. Por esta razão, a estratégia a ele aplicada é baseada em um método de teste também estrutural, conhecido como OBT (*Oscillation-Based Test*). Neste método o circuito é dividido em blocos simples que são transformados em osciladores. Os parâmetros do sinal obtido, tais como a frequência de oscilação e a amplitude, têm relação direta com os componentes utilizados na implementação do oscilador. Desta maneira, é possível detectar falhas no FPAA observando os parâmetros do sinal gerado. Esta estratégia é estudada inicialmente considerando uma análise externa dos parâmetros do sinal. Como uma alternativa de redução de custos e melhoria na cobertura de falhas, um analisador de resposta baseado em um duplo integrador é adotado, permitindo que a avaliação do sinal gerado pelo oscilador seja feita internamente, utilizando-se os recursos programáveis do próprio FPAA. Os resultados obtidos para as análises interna e externa são então comparados. O segundo FPAA estudado, da *Anadigm Company*, é um dispositivo a capacitores chaveados que tem como característica a programabilidade funcional. Por esta razão o desenvolvimento de uma técnica de teste estrutural é dificultado, pois não se conhece detalhes da arquitetura do componente. Por esta razão, uma técnica de teste funcional, conhecida como *Transient Response Analysis Method*, é aplicada ao teste deste FPAA. Neste método o circuito sob teste é dividido em blocos funcionais de primeira e segunda ordem e a resposta transiente destes blocos para um dado estímulo de entrada é analisada. O bloco sob teste é então duplicado e um esquema de auto-teste integrado baseado em redundância é desenvolvido, com o intuito de se obter um sinal de erro. Este sinal de erro representa a diferença das respostas transientes dos blocos duplicados. Como proposta para se aumentar a observabilidade do sinal de erro o mesmo é integrado ao longo tempo, aumentando a capacidade de detecção de falhas quando utilizado este método. Em ambas estratégias o objetivo principal do trabalho é testar os blocos analógicos programáveis dos FPAA explorando ao máximo a programabilidade dos dispositivos e utilizando recursos pré-existentes para auxiliar no teste. Os resultados obtidos mostram que as estratégias desenvolvidas configuram boas alternativas para o auto-teste integrado deste tipo de componente.

**Palavras-chave:** Teste Analógico, Auto-Teste Integrado (Embarcado), Field Programmable Analog Array (FPAA), Teste Baseado em Oscilação, Análise de Resposta Transiente.

## ABSTRACT

This work addresses the test of programmable analog devices. Several analog test methodologies are studied and some of them are applied in the developed strategies. In order to validate these strategies, two commercial FPAA (Field Programmable Analog Arrays), of different vendors and distinct models, are considered as devices under test. The first studied device is a continuous-time FPAA from *Lattice Semiconductors*. One important characteristic of such device is the structural programmability. For this reason the test strategy applied to this FPAA is based in a structural method known as OBT (Oscillation-Based Test). In this method, blocks of the circuit under test are individually converted into oscillators. The parameters of the generated signal, such as the frequency and amplitude, can be expressed as function of the components used in the oscillator implementation. This way, it is possible to detect faults in the FPAA simply observing such parameters. This method is firstly studied considering an external analysis of the signal parameters. However, in a second moment, an internal response analyzer, based on a double integrator, is built with the available programmable resources of the FPAA. This way, overall test cost is reduced, while the fault coverage is increased with no area overhead. The obtained results considering the external analysis and the built-in response evaluation are compared. The second considered FPAA, from *Anadigm Company*, is a switched capacitor device whose programming characteristic is strictly functional. Thus, a structural test method cannot be easily developed and applied without the previous knowledge of the device architectural details. For this reason, a functional test method known as TRAM (Transient Response Analysis Method) is adopted. In this method the Circuit Under Test (CUT) is programmed to implement first and second order blocks and the transient response of these blocks for a given input stimuli is analyzed. Taking advantage of the inherent programmability of the FPAA, a BIST-based scheme is used in order to obtain an error signal representing the difference between the fault-free and faulty Configurable Analog Blocks (CABs). As a proposal to augmenting the observability, the error signal is integrated, enhancing the fault detection capability when using this method. In both developed strategies the main objective is to test the CABs of the FPAA exploiting the device programmability, using the existing resources in order to aid the test. The obtained results show that the developed strategies represent good alternatives to the built-in self-test of such type of device.

**Keywords: Analog Test, Built-In Self-Test, Field Programmable Analog Array (FPAA), Oscillation-based Test, Transient Response Analysis Method.**

## SUMÁRIO

<b>1. INTRODUÇÃO.....</b>	<b>14</b>
<b>2. TESTE DE CIRCUITOS ANALÓGICOS – O ESTADO DA ARTE .....</b>	<b>18</b>
2.1 <i>Teste em produção .....</i>	<i>19</i>
2.2 <i>Teste Funcional x Teste Estrutural .....</i>	<i>19</i>
2.3 <i>Auto-Teste Integrado e Projeto Visando a Testabilidade.....</i>	<i>21</i>
2.4 <i>Modelos de Falhas para o Teste Analógico.....</i>	<i>22</i>
2.5 <i>Teste em Funcionamento (on-line).....</i>	<i>23</i>
2.6 <i>O Padrão IEEE 1149.4.....</i>	<i>27</i>
2.7 <i>Analog Built-in Block Observer (ABILBO) .....</i>	<i>29</i>
2.8 <i>Teste Baseado em Oscilação .....</i>	<i>30</i>
2.9 <i>Método De Teste Baseado na Análise de Resposta Transiente.....</i>	<i>34</i>
2.10 <i>Teste de FPAAs.....</i>	<i>38</i>
<b>3. FIELD PROGRAMMABLE ANALOG ARRAYS (FPAAS) .....</b>	<b>40</b>
3.1 <i>Arquitetura típica de um FPAA genérico .....</i>	<i>40</i>
3.2 <i>FPAAs Acadêmicos .....</i>	<i>42</i>
3.3 <i>FPAAs Comerciais.....</i>	<i>46</i>
3.3.1 <i>A Família ispPAC da Lattice Semiconductors .....</i>	<i>50</i>
3.3.2 <i>O FPAA AN10E40 da Anadigm Company .....</i>	<i>53</i>
3.3.3 <i>O FPAA AN221E04 da Anadigm Company .....</i>	<i>57</i>
<b>4. TESTE DO FPAA ISPPAC10 DA LATTICE SEMICONDUCTORS .....</b>	<b>63</b>
4.1 <i>OBT Associado à Análise Externa: BIST Parcial.....</i>	<i>63</i>
4.1.1 <i>Teste dos Capacitores Programáveis.....</i>	<i>66</i>
4.1.2 <i>Teste dos Blocos de Ganho (IAs).....</i>	<i>69</i>
4.1.3 <i>Teste das Chaves Programáveis dos CABs .....</i>	<i>70</i>
4.2 <i>OBT Associado à Análise Interna de Resposta: BIST Completo .....</i>	<i>72</i>
4.2.1 <i>Análise de Resposta para o Teste dos Capacitores Programáveis .....</i>	<i>74</i>
4.2.2 <i>Análise de Resposta para o Teste dos Blocos de Ganho (IAs).....</i>	<i>79</i>
4.2.3 <i>Análise de Resposta para o Teste das Chaves Programáveis dos CABs .....</i>	<i>81</i>
4.2 <i>Avaliação dos Resultados .....</i>	<i>83</i>
<b>5. TESTE DO FPAA AN221E04 DA ANADIGM COMPANY .....</b>	<b>86</b>
5.1 <i>Análise de Sensibilidade dos Parâmetros da Resposta Transiente.....</i>	<i>86</i>

<b>5.2 Efeitos da Transformação em Frequência</b> .....	<b>93</b>
<b>5.3 Aplicação do Teste</b> .....	<b>95</b>
<b>5.4 Injeção de Falhas e Simulação</b> .....	<b>98</b>
<b>5.5 Resultados Experimentais</b> .....	<b>100</b>
5.5.1 Análise de Sensibilidade Dos Parâmetros Funcionais do Circuito .....	100
5.5.2 Estimativa da Cobertura de Falhas e Tempo de Teste.....	105
<b>6. CONCLUSÕES</b> .....	<b>108</b>
<b>APÊNDICE A:</b> Código em <i>Matlab</i> para cálculo da sensibilidade dos parâmetros da resposta ao step em relação aos parâmetros das funções de transferência de 1 <sup>a</sup> e 2 <sup>a</sup> ordem.....	<b>118</b>
<b>APÊNDICE B:</b> Código em <i>Matlab</i> para cálculo da sensibilidade dos parâmetros das funções de transferência de 1 <sup>a</sup> e 2 <sup>a</sup> ordem a variações nos capacitores programáveis dos CABs .....	<b>120</b>
<b>ANEXO A:</b> Características Elétricas do FPAA ispPAC10 da <i>Lattice</i> .....	<b>124</b>
<b>ANEXO B:</b> Características Elétricas do FPAA AN10E40 da <i>Anadigm</i> .....	<b>125</b>
<b>ANEXO C:</b> Características Elétricas do FPAA AN221E04 da <i>Anadigm</i> .....	<b>126</b>
<b>ANEXO D:</b> Comparativo das Características Gerais dos FPAA's da Família <i>Vortex</i> da <i>Anadigm</i> .....	<b>127</b>

## LISTA DE ILUSTRAÇÕES

Figura 2.1: Estimativa de custo relativo por falha em cada etapa da produção de um sistema eletrônico .....	19
Figura 2.2: Diagrama em blocos de um esquema de auto-teste completo .....	22
Figura 2.3: Votador analógico e esquemático dos blocos K e $\Sigma$ . Extraído de ( VÁSQUEZ; RUEDA; HUERTAS, 1991).....	25
Figura 2.4: Estrutura global de teste compatível com o padrão IEEE 1149.4. Extraída de (OSSEIRAN, 1999) .....	28
Figura 2.5: Arquitetura de um ABM. Extraída de (OSSEIRAN, 1999).....	28
Figura 2.6: Analisador de resposta baseado em duplo integrador.....	29
Figura 2.7: Efeito da dupla integração sobre um sinal senoidal.....	30
Figura 2.8: Estrutura simplificada do teste baseado em oscilação. Adaptada de (ARABI; KAMINSKA, 1996 ).....	31
Figura 2.9: Representação em blocos de um sistema realimentado e sua função de transferência.....	32
Figura 2.10: Transformação do bloco sob teste em um oscilador utilizando um elemento de realimentação não linear .....	33
Figura 2.11: (a) Esquema de avaliação intrachip. (b) Formas de onda. Extraído de (VÁSQUEZ <i>et al.</i> , 2002).....	34
Figura 2.12: Resposta ao <i>step</i> de um filtro passa-baixa de segunda ordem .....	35
Figura 2.13: Respostas ao <i>step</i> de um filtro passa-baixa de segunda ordem com desvios de +20% nos parâmetros da função de transferência .....	36
Figura 2.14: Resposta ao <i>step</i> de um sistema de primeira ordem .....	36
Figura 2.15: Transformação em frequência através da integração do estímulo de entrada.....	38
Figura 3.1: Arquitetura de um FPAA genérico .....	41
Figura 3.2: Arquitetura de um CAB genérico .....	41
Figura 3.3: (a) Célula configurável e (b) elementos primitivos da arquitetura proposta em (LEE; GULAK, 1991) .....	42
Figura 3.4: Arquitetura do FPAA proposto em (LEE; GULAK, 1995).....	43
Figura 3.5: Arquitetura do FPAA proposto em (LEE; HUI, 1998).....	44
Figura 3.6: (a) Arquitetura do CAB e (b) do banco de capacitores do FPAA proposto em (LEE; HUI, 1998) .....	44
Figura 3.7: CAB do FPAA proposto em (KUTUK; KANG, 1998).....	45
Figura 3.8: Bloco de interconexão do FPAA proposto em (KUTUK; KANG, 1998).....	45
Figura 3.9: (a) Esquemático e (b) simbologia do carregador de corrente segundo (PREMONT, 1998).....	46
Figura 3.10: Funções analógicas implementáveis no TRAC20LH (ZETEX, 1999).....	47
Figura 3.11: Estrutura global do FPAA TRAC20LH da Zetex (ZETEX, 1999) .....	47
Figura 3.12: Representação em blocos do PSoC (CYPRESS, 2002).....	48
Figura 3.13: Esquemático da célula analógica de tempo contínuo do PSoC (CYPRESS, 2002).....	49
Figura 3.14: Esquemático da célula analógica a capacitores chaveados do PSoC (CYPRESS, 2002) .....	49

Figura 3.15: Representação em blocos dos componentes da família ispPAC da Lattice (LATTICE, 2000).....	50
Figura 3.16: Esquemático do CAB dos componentes da família ispPAC (LATTICE, 2000) ...	51
Figura 3.17: Representação do CAB dos componentes da família ispPAC (LATTICE, 2000) .....	51
Figura 3.18: Representação global do FPAA ispPAC10 (LATTICE, 2000) .....	52
Figura 3.19: Resultado de uma simulação para um filtro programado com frequência de corte igual a 595KHz .....	53
Figura 3.20: Representação em blocos do AN10E40 (ANADIGM, 2003).....	54
Figura 3.21 Barramentos globais e <i>cross over switches</i> .....	55
Figura 3.22: Conectividade local do AN10E40 (BRATT; MACBETH, 1998).....	55
Figura 3.23: Representação do CAB do AN10E40 (ANADIGM, 2003).....	56
Figura 3.24: Banco programável de capacitores .....	57
Figura 3.25: Diagrama em blocos do FPAA AN221E04. Extraído de (ANADIGM, 2003b) ...	58
Figura 3.26: Esquemático do CAB do FPAA AN221E04. Extraído de (ANADIGM, 2003b)..	59
Figura 3.27: Esquemático considerado para os bancos de capacitores do AN221E04.....	60
Figura 3.28(a): Célula de entrada do FPAA AN221E04. Extraído de (ANADIGM, 2003b) ....	60
Figura 3.28(b): Célula de entrada multiplexada do FPAA AN221E04. Extraído de (ANADIGM, 2003b).....	60
Figura 3.29: Célula de saída do FPAA AN221E04. Extraído de (ANADIGM, 2003b) .....	61
Figura 3.30: Exemplo de programabilidade com o software <i>Anadigm Designer II</i> .....	61
Figura 4.1: Esquemático do oscilador utilizado neste trabalho .....	63
Figura 4.2: Pólos do oscilador localizados sobre o eixo $j\omega$ .....	65
Figura 4.3: Representação dos osciladores na área de trabalho do software <i>PAC-Designer</i> ....	66
Figura 4.4: Frequência de oscilação para o circuito sem falhas e considerando desvios de $\pm 20\%$ nos capacitores programáveis.....	68
Figura 4.5 Amplitude de oscilação para o circuito sem falhas e considerando desvios de $\pm 20\%$ nos capacitores programáveis.....	68
Figura 4.6: Comportamento da frequência em relação ao ganho para os circuitos com e sem falha .....	70
Figura 4.7: Chaves programáveis do CAB do ispPAC10 .....	71
Figura 4.8: Representação do oscilador e do ORA baseado em duplo integrador.....	73
Figura 4.9: Sinal parabólico observado na saída do ORA (integradores puros) em três ensaios para uma mesma configuração.....	73
Figura 4.10: Sinais adquiridos durante ensaios de teste utilizando um integrador com perdas no ORA .....	74
Figura 4.11: Assinaturas de tempo sem falhas e para um desvio de $+20\%$ no capacitor $C_1$ do oscilador.....	76
Figura 4.12: Histograma das diferenças entre as assinaturas de tempo para os circuitos com e sem falhas nos capacitores programáveis considerando os valores menores que 30pF ...	76
Figura 4.13: Assinaturas com e sem falhas no capacitor $C_1$ do oscilador para valores de capacitâncias maiores que 30pF.....	77
Figura 4.14: Assinaturas com e sem falhas no capacitor $C_2$ do oscilador para valores de capacitâncias maiores que 30pF.....	78
Figura 4.15: Assinaturas com e sem falhas no capacitor $C_3$ do oscilador para valores de capacitâncias maiores que 30pF.....	78
Figura 4.16: Assinaturas com e sem falhas no capacitor $C_4$ do oscilador para valores de capacitâncias maiores que 30pF.....	79
Figura 4.17: Assinaturas com e sem falhas para o teste do bloco de ganho $IA_1$ .....	81

Figura 4.18: Histograma das diferenças entre as assinaturas de tempo para os circuitos com e sem falhas nos blocos de ganho segundo o modelo $K_{falha} = K \pm 1$ .....	81
Figura 4.19: Utilização de dois IAs como entrada dos blocos do oscilador.....	82
Figura 5.1: Sensibilidade do parâmetro da resposta ao impulso “Overshoot” a variações no fator de qualidade de um filtro de segunda ordem.....	91
Figura 5.2: Sensibilidade do parâmetro da resposta ao impulso “Tempo de pico” a variações no fator de qualidade de um filtro de segunda ordem.....	92
Figura 5.3: Diagrama de bode de um filtro passa-alta cuja frequência de corte é aproximadamente 125Krad/s .....	94
Figura 5.4: Diagrama de bode de um filtro passa-baixa resultante da transformação em frequência do filtro passa-alta cujo diagrama de bode é mostrado na Figura 5.3.....	94
Figura 5.5: Esquemático do filtro passa-baixa de 1. <sup>a</sup> ordem utilizado .....	96
Figura 5.6: Esquemático do filtro passa-baixa de 2. <sup>a</sup> ordem utilizado .....	97
Figura 5.7: Diagrama em blocos da estrutura de teste considerando um filtro de segunda ordem .....	97
Figura 5.8: Área de trabalho do software <i>Anadigm Designer II</i> com a representação em blocos do circuito de teste programado .....	98
Figura 5.9: Sinal de Erro não integrado e saída dos comparadores para um desvio injetado de +5% na frequência de corte do bloco de 1. <sup>a</sup> ordem.....	99
Figura 5.10: Sinal de Erro não integrado e saída dos comparadores para um desvio injetado de +5% na frequência de corte do bloco de 2. <sup>a</sup> ordem.....	100
Figura 5.11: Sinal de Erro não integrado e saída dos comparadores para um desvio injetado de +5% no fator de qualidade do bloco de 2. <sup>a</sup> ordem .....	100
Figura 5.12: Formas de onda adquiridas durante um dos ensaios de injeção de falhas .....	103

## LISTA DE TABELAS

Tabela 1: Configurações para o teste das chaves .....	71
Tabela 2: Cobertura de falhas e número de configurações para o teste dos capacitores utilizando o ORA .....	79
Tabela 3: Número de configurações e cobertura de falhas obtida para o método OBT com análise externa (OBT) e com análise interna de assinatura (ORA) .....	83
Tabela 4: Resultado da análise de sensibilidade em relação aos parâmetros “frequência de corte” e “ganho” para blocos de 1 <sup>a</sup> ordem .....	90
Tabela 5: Resultado da análise de sensibilidade em relação aos parâmetros “frequência de corte” e “ganho” para blocos de 2 <sup>a</sup> ordem .....	90
Tabela 6: Resultados da análise de sensibilidade para o filtro de 1 <sup>a</sup> ordem .....	104
Tabela 7: Resultados da análise de sensibilidade para o filtro de 2 <sup>a</sup> ordem .....	104
Tabela 8: Variações mínimas detectáveis nos valores dos capacitores dos CABs.....	104

## LISTA DE ABREVIATURAS

ABM: Analog Boundary Module  
A/D: Analog-to-Digital  
ATE: Automatic Test Equipment  
BIST: Built-in Self-Test  
CI: Circuito Integrado  
CMOS: Complementary Metal-Oxide-Semiconductor  
CUT: Circuit Under Test  
D/A: Digital-to-Analog  
DC: Direct Current  
DfT: Design for Testability  
DNL: Differential Non-Linearity  
DSP: Digital Signal Processing  
EEPROM Electrical Erasable Programmable Read Only Memory  
ENoB: Effective Number of Bits  
FFT: Fast Fourier Transform  
IEEE: Institute of Electrical and Electronic Engineers  
INL: Integral Non-Linearity  
IA: Input Amplifier  
I/O: Input/Output  
IP: Intellectual Property  
LUT: Look-Up Table  
OA: Output Amplifier  
OBIST: Oscillation Built-in Self-Test  
OBT: Oscillation-based Test  
OPAMP: Operational Amplifier  
ORA: Output Response Analyzer  
OSR: Over Sample Ratio  
OTS: Oscillation Test Strategy  
PSD: Power Spectrum Density  
RF: Rádio-freqüência  
SAR: Successive Approximation Register  
SFDR: Spurious Free Dynamic Range  
SINAD: Signal-to-Noise-and-Distortion Ratio  
SNR: Signal-to-Noise Ratio  
SoC: System-on-a-Chip  
S-RAM: Static Random Access Memory  
TC: Test Configuration  
THD: Total Harmonic Distortion  
TMR: Triple Modular Redundancy  
TRAM: Transient Response Analysis Method  
XOR: Exclusive OR

## 1. INTRODUÇÃO

Ao longo dos últimos trinta anos as indústrias eletrônica e de semicondutores vêm caminhando a passos largos, evoluindo constantemente e concebendo produtos cada vez mais complexos, versáteis, confiáveis e de melhor desempenho. Uma característica marcante desta notável evolução é a miniaturização, que pode ser medida tomando como exemplo o telefone celular. Este, no início de sua popularização, era um aparelho pesado, grande e cuja bateria poderia durar menos de 24 horas. Já os modernos aparelhos disponíveis atualmente cabem na palma da mão, são leves e tem autonomia energética de vários dias, sem falar em suas inúmeras funcionalidades que vão muito além da possibilidade de se efetuar ligações telefônicas.

Este aumento de desempenho, menor consumo de energia e diminuição nas dimensões físicas, encontram suporte na indústria microeletrônica. A complexidade dos CIs (Circuitos Integrados) aumenta ao passo que as dimensões mínimas dos transistores diminuem, permitindo assim, que um único circuito integrado seja composto por milhões de transistores.

O aumento da capacidade de integração gera um aumento na produtividade (*yield*) e na velocidade de operação dos CIs, permitindo a concepção de circuitos mais baratos (custo relativo por transistor) e capazes de operar em frequências mais elevadas. Em contrapartida, os circuitos se tornam mais susceptíveis a falhas, pois, com o tamanho reduzido dos transistores as impurezas presentes no silício (ou ali depositadas involuntariamente durante o processo de produção) podem ser fisicamente maiores do que um ou mais transistores, podendo danificar o circuito (LEHTONEN; POSILA; ISOAHO, 2005). A diminuição das dimensões dos transistores também aumenta sua susceptibilidade às variações do processo de produção, o que pode resultar em circuitos operando fora de suas faixas de especificação. Outros efeitos colaterais da miniaturização são: o efeito de canal curto, a corrente de fuga (*leakage current*) (TAUR *et al.*, 1996), e uma maior vulnerabilidade a falhas transientes ocasionadas por incidência de radiação (MESSENGER, 1992). Assim, ao passo que aumenta a complexidade dos circuitos integrados, aumenta também a complexidade do projeto e do teste destes circuitos.

A lei de Moore (MOORE, 1965) para circuitos digitais postula que a cada 18 meses a capacidade de integração de transistores em um circuito integrado digital é multiplicada por dois. O mesmo não acontece com circuitos analógicos, fazendo com que, quando possível, os circuitos digitais sejam preferidos na implementação de sistemas eletrônicos. No entanto, em

muitas aplicações os circuitos analógicos desempenham um importante papel, principalmente no interfaceamento do sistema digital com o mundo físico. O mundo real é, por natureza, essencialmente analógico. Portanto, a tarefa de interfaceamento não pode ser executada por circuitos puramente digitais. Filtros contínuos e chaveados, comparadores de tensão e amplificadores são circuitos analógicos muito utilizados na implementação de sistemas eletrônicos.

A necessidade de utilização de circuitos analógicos impõe requisitos como confiabilidade, precisão e operação segura, o que há mais tempo já vinha sendo preocupação no domínio digital. Por esta razão, nas últimas duas décadas, o teste de circuitos analógicos vem recebendo especial atenção tanto da indústria como da comunidade científica, o que pode ser mensurado pelo crescente número de trabalhos publicados nesta área.

Recentemente, o surgimento de um novo tipo de circuito analógico, denominado *Field Programmable Analog Array* (FPAA), trouxe para o domínio analógico características até então restritas ao domínio digital, proporcionadas pelos *Field Programmable Gate Arrays* (FPGAs). Tais características são a reconfigurabilidade, flexibilização de projeto e prototipação rápida (HEREFORD; PRUITT, 2004; ZNAMIROWSKI; PAULUSINSKI; VRUDHULA, 2004). Uma vez que os FPAA's representam uma boa alternativa ao projeto de circuitos analógicos, o teste deste tipo de dispositivo ganha importância, pois é preciso garantir aos FPAA's os requisitos de confiabilidade necessários aos sistemas eletrônicos atuais.

O teste de circuitos analógicos apresenta uma série de dificuldades, entre elas a falta de padronização (diferentes tipos de circuitos requerem técnicas e metodologias distintas) e a dificuldade de geração de vetores de teste, uma vez que os sinais considerados são contínuos e podem assumir infinitos valores mesmo considerando intervalos finitos. O desafio de testar circuitos programáveis é ainda maior, pois a programabilidade permite que um dado circuito sob teste tenha os valores de seus componentes modificados ao longo de uma ampla faixa. Isto, em geral, acarreta na necessidade de se repetir o teste de uma dada topologia diversas vezes, a fim de contemplar toda a faixa programável dos componentes que se deseja testar.

Este trabalho visa desenvolver e aplicar técnicas e estratégias de teste para dispositivos analógicos programáveis (FPAA's). Como veículos de teste são utilizados dois dispositivos de diferentes fabricantes. Cada um dos FPAA's apresenta características distintas quanto a programabilidade e funcionalidade, logo, diferentes metodologias e abordagens são empregadas no teste destes dispositivos.

Ao FPAA cuja abordagem de programação é estrutural é aplicado um método de teste orientado a defeitos, conhecido como OBT (*Oscillation-Based Test*, ou teste baseado em oscilação) (ARABI; KAMINSKA, 1996). Em tal método o circuito sob teste (ou CUT, do inglês *Circuit Under Test*) é programado para operar como um oscilador e os parâmetros de saída do circuito (frequência e amplitude do sinal oscilatório) são dependentes dos componentes do circuito. Assim, uma falha em um destes componentes pode ser detectada através da observação destes parâmetros. Nesta etapa do trabalho, além de programar parte do FPAA como oscilador, parte do dispositivo é programada como um analisador de resposta. Desta maneira, os recursos externos necessários ao teste são simplificados, o que permite a utilização de um testador externo de muito baixo custo.

O segundo FPAA considerado neste trabalho tem como característica uma programabilidade estritamente funcional, ou seja, o usuário pode modificar tipos e parâmetros de funções de transferência, mas não há acesso direto aos valores dos componentes de um bloco analógico programável. Para este dispositivo é aplicada uma técnica de teste conhecida como *Transient Response Analysis Method* (TRAM, ou teste baseado em análise de resposta transiente) (CALVANO; ALVES, LUBASZEWSKI, 1999). A técnica de análise de transiente visa justamente detectar desvios em parâmetros de funções de transferência de 1<sup>a</sup> e 2<sup>a</sup> ordem, tendo sido aplicada com sucesso no teste de filtros, detectando falhas em parâmetros como ganho, frequência de corte e fator de qualidade. No desenvolvimento desta parte do trabalho uma análise de diferentes funções de transferência é realizada, a fim de se obter uma quantização da sensibilidade dos parâmetros da resposta transiente em relação aos parâmetros funcionais do bloco em questão. Posteriormente, uma análise de sensibilidade é feita sobre a variação dos parâmetros funcionais do bloco em relação a variações nos componentes físicos utilizados na sintetização deste bloco. O objetivo destas análises é identificar uma função de transferência realizável que apresente a maior sensibilidade possível em relação à resposta transitória, e, posteriormente, implementá-la no FPAA de maneira a maximizar o número de componentes testados bem como a sensibilidade dos parâmetros funcionais em relação a estes componentes.

As duas estratégias desenvolvidas neste trabalho têm como objetivo a utilização otimizada dos recursos programáveis dos FPAA's para auxiliar no teste, configurando assim metodologias de auto-teste integrado ou BIST (do inglês *Built-In Self-Test*). Assim, deseja-se obter estratégias de teste de baixo custo e com boa cobertura de falhas quando aplicadas aos

dispositivos considerados neste trabalho. Porém, espera-se que estas estratégias possam ser estendidas a outros modelos de FPAA.

Este trabalho está organizado da seguinte maneira: no capítulo 2 uma revisão sobre metodologias de teste analógico é feita. No capítulo 3 a arquitetura de um FPAA genérico é descrita e diversos FPAAs, acadêmicos e comerciais, são apresentados. O capítulo 4 apresenta o teste do FPAA ispPAC10 através do método OBT em duas abordagens: análise externa e interna dos padrões de teste. O capítulo 5 mostra a aplicação do método TRAM ao FPAA AN221E04. Finalmente, o capítulo 6 apresenta algumas conclusões e sugestões de trabalhos futuros.

## 2. TESTE DE CIRCUITOS ANALÓGICOS – O ESTADO DA ARTE

Diferentemente dos circuitos digitais, que possuem um alto grau de padronização do teste, as partes analógicas demandam diferentes metodologias de teste e modelos de falhas, dependendo do circuito considerado e dos requisitos que se deseja cumprir. Geralmente as especificações de circuitos analógicos são extremamente amplas, uma vez que os valores de frequência e amplitude de um sinal analógico podem assumir infinitos valores. A natureza e o tipo de aplicação de um circuito analógico têm relação direta com a metodologia de teste mais adequada a ser aplicada, o que por sua vez influencia nas medidas de testabilidade como a cobertura de falhas e o tempo de teste (SUNTER; NAGI, 1999). O tempo de teste é um fator que influencia fortemente o custo do teste e conseqüentemente o custo do circuito final (SOUDERS; STENBAKKEN, 1990), uma vez que os testadores (principalmente os analógicos com faixa de frequência ampla) são equipamentos caros, e, quanto mais demorado é o teste, menor é a amortização do custo global do teste sobre o circuito final.

Uma medida da eficiência do teste pode ser tomada utilizando dois parâmetros relativos ao teste: tempo de teste e cobertura de falhas. Espera-se que o teste de um determinado circuito seja suficientemente breve para não se tornar o gargalo da linha de produção e ao mesmo tempo deseja-se que a maior parte das falhas (cuja ocorrência seja mais provável) seja detectada.

Neste sentido muito esforço vem sendo realizado com a finalidade de desenvolver metodologias que facilitem o teste de circuitos analógicos, diminuindo os requisitos de complexidade ou até mesmo eliminando a necessidade de testadores automatizados, o que por sua vez reduz drasticamente o custo do teste.

Neste capítulo é feita uma revisão sobre algumas das técnicas de teste analógicos mais utilizadas ao longo dos últimos anos iniciando com a definição de alguns conceitos que envolvem o teste de sistemas eletrônicos dando especial atenção ao teste de circuitos analógicos.

As seções 2.7, 2.8, e 2.9 são dedicadas à apresentação de três técnicas utilizadas neste trabalho: BIST baseado em dupla Integração (ABILBO: *Analog Built-In Block Observer*), teste baseado em oscilação (OBT) e teste funcional baseado em análise de resposta transiente (TRAM: *Transient response Analysis Method*).

## 2.1 TESTE EM PRODUÇÃO

O teste em produção visa detectar (o mais cedo possível) falhas que possam invalidar o circuito integrado que está sendo fabricado. Há uma teoria informal que diz que o custo do reparo da falha é multiplicado por dez em cada nível de produção de um sistema eletrônico, conforme a Figura 2.1. Por isto, a primeira etapa do teste, após a concepção do CI, é feita sobre o *waffer* de silício, antes mesmo do encapsulamento do circuito. A finalidade deste teste é descartar da linha de produção os circuitos integrados (*dies*) que não passarem no teste, por isso esta modalidade de teste é também conhecida como “*go/no go*” (MILOR; VISVANATHAN, 1987). Outra finalidade do teste em produção é realimentar o processo global de projeto, identificando pontos que podem conter erros ou prejudicar o correto funcionamento do circuito, para que através de mudanças no projeto ou no processo de produção estas deficiências sejam sanadas ou reduzidas.

O teste em produção é feito utilizando equipamentos específicos, os ATEs (*Automatic Test Equipments*) que em geral são equipamentos caros. Por esta razão, o tempo de teste é um fator determinante no custo deste, pois quanto menor é este tempo maior é o volume de CIs testados em um determinado período de tempo e maior é a amortização do custo do teste (que é responsável por boa parte do custo do circuito final) (SOUDERS; STENBAKKEN, 1990).

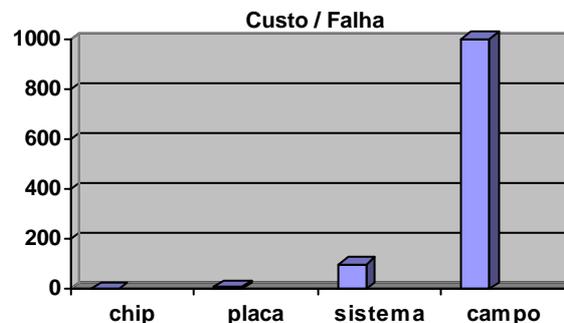


Figura 2.1: Estimativa de custo relativo por falha em cada etapa da produção de um sistema eletrônico

## 2.2 TESTE FUNCIONAL X TESTE ESTRUTURAL

Quanto aos objetivos do teste, em uma visão geral, pode-se classificá-lo como teste funcional e teste estrutural. O teste funcional visa identificar se o projeto está de acordo com as especificações e se o circuito tem um comportamento funcional correto (LUBASZEWSKI; COTA; KRUG, 2000). Por isto o teste funcional é também conhecido como teste baseado em

especificações (*specification-based test*). Muitas vezes utiliza-se como vetores de teste funcional os mesmos estímulos considerados pelos projetistas para a validação e desenvolvimento do projeto durante as simulações. Desta forma o custo do teste se reduz ao custo de aplicação do mesmo, uma vez que a etapa de geração do teste já foi cumprida *a priori* (RENOVELL, 2004).

O teste estrutural deve verificar defeitos de fabricação que façam com que o circuito obtido seja estruturalmente diferente do circuito sem falhas. Defeitos estruturais podem ser causados por impurezas presentes ou depositadas no silício ou nas máscaras (*spot-defect*), desalinhamento das máscaras e falhas nos processos de corrosão e implantação de metal e óxido. Tais defeitos usualmente causam falhas catastróficas (curto-circuito e circuito-aberto). Por isso diz-se que o teste estrutural é orientado a falhas ou orientado a defeitos (RENOVELL, 2004).

Em geral, se os detalhes de implementação, as características funcionais e a performance do circuito forem conhecidas é possível aplicar tanto um teste funcional como estrutural. Uma abordagem híbrida também pode ser adotada, mesclando o teste funcional com o teste estrutural com o intuito de reduzir o tempo ou custo de aplicação do teste (BELL; SPINKS, 1998).

Como um exemplo da aplicação das duas diferentes abordagens é possível citar o teste de conversores de dados (Analógicos para digital - A/D e digital para analógico - D/A). Os conversores de dados são componentes mistos, ou seja, mesclam características dos domínios digital e analógico. Tais componentes são de crucial importância para os sistemas eletrônicos, pois atuam nas interfaces entre o mundo físico (analógico) e mundo computacional (digital). Dado o grande volume de produção deste tipo de componente o teste torna-se uma etapa decisiva e custosa, pois os conversores de dados atuais são componentes de alto desempenho, o que aumenta os requisitos do teste.

O teste funcional de conversores de dados visa avaliar características estáticas e dinâmicas. Entre as características estáticas dos conversores estão os erros de não linearidade integral e diferencial (INL e DNL), ganho e *offset*. Tais características podem ser avaliadas através do método do Histograma Linear (DOERNBERG, LEE, HODGES, 1984). Métodos baseados em análise espectral (FTT: *Fast Fourier Transform*) podem ser utilizados para avaliar características funcionais dinâmicas como a relação sinal ruído e distorção (SINAD: *Signal-to-Noise-and-Distortion Ratio*), distorção harmônica total (THD: *Total Harmonic*

*Distortion*), a faixa dinâmica livre de componentes espúrios (SFDR: *Spurious Free Dinamic Range*) e o número efetivo de BITS (ENoB: *Effective Number of BITS*) (FLORES, 2003).

O teste funcional de conversores de dados pode ser computacionalmente custoso, dependendo das características do componente. Em (PERALÍAS; RUEDA; HUERTAS, 2001) duas estratégias de auto-teste integrado estrutural para conversores A/D do tipo *pipelined* foram propostas. Segundo os autores os métodos propostos requerem estímulos de teste de fácil geração e apresentam bons resultados.

As abordagens de teste estrutural e funcional podem ser aplicadas a praticamente todos o tipos de circuitos analógicos, dependendo do nível de abstração do modelo de falhas e dos objetivos do teste em si. Neste trabalho as duas estratégias de teste desenvolvidas para os diferentes modelos de FPAA's diferem quanto ao tipo de teste aplicado. Uma abordagem estrutural, orientada a defeitos, foi empregada no teste de um FPAA cujos detalhes estruturais são conhecidos. A segunda estratégia é derivada de uma metodologia de teste funcional, baseada em especificações de parâmetros de alto nível. A aplicação de um método funcional se fez necessária devido à falta de informações sobre detalhes da implementação estrutural do componente considerado. As metodologias utilizadas nas duas estratégias são apresentadas com detalhes nos capítulos 4 e 5.

### **2.3 AUTO-TESTE INTEGRADO E PROJETO VISANDO A TESTABILIDADE**

Assim como no domínio do teste digital, o teste analógico segue uma tendência de integrar ao projeto do circuito-alvo recursos que auxiliem o teste, tais como geradores de estímulo e analisadores de resposta, ou simplesmente prover mecanismos de acesso a pontos internos do circuito. A estratégia de modificar o circuito original adicionando componentes (que não são necessários ao funcionamento do circuito) para auxiliar no teste é denominado Projeto Visando a Testabilidade (DfT: *Design for Testability*). Já a técnica que utiliza recursos internos previamente existentes ou inseridos por meio de DfT para realizar ou auxiliar o teste recebe o nome de Auto-Teste Integrado (BIST: *Built-In Self-Test*) (CHATTERJEE; NAGI, 1996). A Figura 2.2 exemplifica a técnica de BIST considerando um auto-teste integrado total, ou seja, tanto a geração de estímulos quanto a análise de resposta são feitas internamente. Algumas técnicas propõem a reconfiguração de blocos analógicos do próprio circuito para operar como geradores de estímulo de teste (MIR; LUBASZEWSKI; CURTOIS, 1996), analisadores de resposta ou ambos (LUBASZESWKI; MIR; PULZ, 1996).

A utilização de técnicas de *BIST* diminui os requisitos de complexidade dos testadores externos, diminuindo também o custo e em algumas vezes até mesmo o tempo de teste. Em contrapartida, tais técnicas usualmente penalizam o circuito com um aumento de área (*overhead*) causando um decréscimo na produtividade (*yield*). Outra restrição do auto-teste integrado é a dissipação de potência durante o teste, pois esta não pode exceder os limites máximos suportados pelo componente. E finalmente, em se tratando de circuitos analógicos, a inserção de elementos extras (principalmente chaves no caminho do sinal) pode degradar a performance do circuito. Porém, ainda assim, as técnicas baseadas em *BIST* são muito atrativas, pois usualmente proporcionam um baixo custo de teste mesmo acarretando em perda de performance. Neste sentido, existe um compromisso entre custo do teste e performance do circuito que precisa ser corretamente equalizado na busca de um resultado satisfatório.

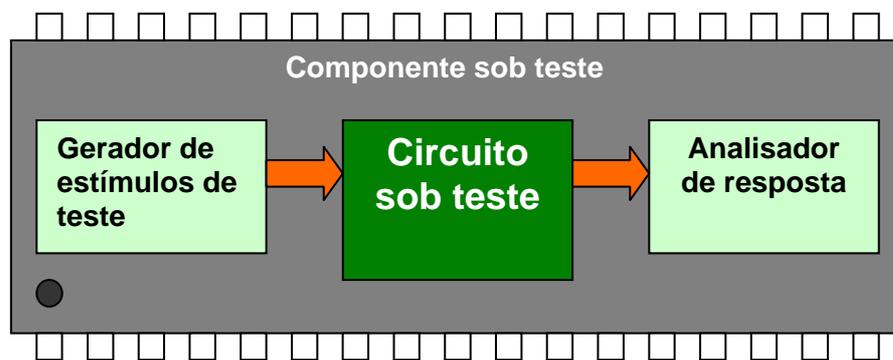


Figura 2.2: Diagrama em blocos de um esquema de auto-teste completo

## 2.4 MODELOS DE FALHAS PARA O TESTE ANALÓGICO

As diversas técnicas de teste analógico existentes se aplicam especificamente a alguns tipos de circuitos, pois, tais técnicas levam em conta características do circuito como complexidade, requisitos, aplicação, funcionalidade e a maneira como ele foi projetado (SUNTER; NAGI, 1999).

Diferenças também são constantes quanto aos modelos de falhas utilizados em técnicas de teste distintas, aplicadas a diferentes circuitos-alvo. No teste digital existe um modelo de falhas que pode ser aplicado a praticamente todos os circuitos, trata-se do modelo de colagem (*stuck-at*) (HUGHES, 1988) em que um dado nó do circuito tem seu valor lógico sempre “colado” em um dos níveis digitais (0 ou 1). Já nos circuitos analógicos, o modelo de falhas

mais comum representa desvios percentuais dos valores nominais dos componentes do circuito (falhas paramétricas), bem como falhas de curto-circuito e circuito-aberto (falhas catastróficas) entre componentes e conexões (SOMA, 1996). Entretanto os valores considerados destes desvios nominais dependem de fatores inerentes ao circuito em questão, como a tolerância dos componentes e a sensibilidade de parâmetros funcionais a desvios nestes componentes. Outro modelo de falhas que também pode ser adotado leva em conta justamente os parâmetros funcionais do circuito. Em um modelo de falhas funcional as falhas são representadas por desvios em parâmetros que representam o comportamento do circuito, como, por exemplo, ganho e frequência de corte de filtros.

Falhas paramétricas são usualmente ocasionadas por variações estatísticas em parâmetros do processo de produção (medidas em termos do desvio padrão,  $\sigma$ ) tais como a espessura de óxido, dopagem, espessura das linhas de metal, polysilício e difusão, bem como o desalinhamento de máscaras (SUNTER; NAGI, 1999). Falhas catastróficas podem ocorrer devido a impurezas no silício ou nas máscaras, bem como falhas no processo de produção (corrosão e deposição de metal, por exemplo). Estudos realizados sobre a ocorrência de falhas em circuitos integrados analógicos mostram que cerca de 90% das falhas reportadas são catastróficas, consistindo em curto-circuito ou circuito-aberto em transistores, díodos, capacitores e linhas de interconexão (STAPPER; ARMSTRONG; SAJI, 1983; MILOR; VISVANATHAN, 1989).

## **2.5 TESTE EM FUNCIONAMENTO (*ON-LINE*)**

O teste em funcionamento (*on-line*) visa detectar falhas durante o funcionamento do sistema no qual o circuito sob teste encontra-se inserido. Tal modalidade de teste é importante em sistemas que requerem confiabilidade e operação segura, como, por exemplo, sistemas controladores de aeronaves, trens e usinas nucleares.

Em geral, o teste *on-line* visa apenas detectar uma falha em funcionamento e indicar a presença desta, mas, em alguns casos, mecanismos específicos podem ser utilizados para corrigir esta falha ou compensar desvios em parâmetros importantes do circuito (HAN *et al.*, 2005). Estas falhas podem não ter sido detectadas pelo teste em produção ou podem ter se manifestado posteriormente, devido a fatores como o envelhecimento do circuito, vibração ocasionada durante o transporte, mau uso e interação com o meio.

Assim como no domínio digital o teste *on-line* de circuitos analógicos pode ser baseado em redundância (duplicação ou triplicação do hardware), onde as partes replicadas têm suas saídas comparadas, com o objetivo de detectar desvios comportamentais. Tal técnica é também conhecida como “teste concorrente”.

Em 1991, VÁSQUEZ, RUEDA e HUERTAS propuseram um esquema baseado em redundância modular tripla (TMR: *Triple Modular Redundancy*) com o objetivo de agregar a característica de tolerância a falhas aos circuitos a capacitores chaveados. Em tal esquema o circuito considerado é triplicado e suas saídas alimentam um votador analógico. O votador compara mutuamente as três saídas do circuito visando detectar diferenças entre elas. Para esta finalidade é preciso que tal comparação seja feita de modo “flexível”, pois, devido à natureza analógica dos sinais comparados, é pouco provável que mesmo na ausência de falhas estes sejam exatamente iguais. Assim é preciso considerar uma banda de tolerância dentro da qual os sinais podem diferir sem prejudicar o comportamento do circuito. O resultado desta comparação serve como controle de um somador programável que realiza a média aritmética dos três sinais de saída do circuito triplicado ( $V_1$ ,  $V_2$  e  $V_3$ ) caso não exista nenhuma indicação de diferença mútua entre os sinais. Caso um dos sinais sofra um desvio significativo em relação aos outros dois, dois dos três comparadores irão indicar tal diferença. Assim é possível identificar qual das três partes do circuito não está funcionando corretamente e, através de chaves programáveis, desconectá-la do circuito somador ao mesmo tempo se procede a troca dos seus pesos, a fim de fazer a média aritmética entre os dois sinais provenientes dos circuitos considerados “sem falha”. O esquema global do votador e o esquemático dos blocos comparadores (K) e do somador ( $\Sigma$ ) são mostrados na Figura 2.3. O circuito final é baseado em um esquema de capacitores chaveados controlado por quatro sinais de relógio com fases distintas.

A redundância de hardware implica em um aumento significativo de área ocupada em silício, o que pode ser proibitivo dependendo da aplicação alvo. A escolha por um esquema de redundância depende dos requisitos de confiabilidade necessários ao circuito, pois é preciso justificar o acréscimo de área, o aumento no consumo de energia e o decréscimo na produtividade (*yield*).

Em (HUERTAS; VÁSQUEZ; RUEDA, 1992) os mesmos autores propõem um esquema para diminuir o problema de acréscimo de área no teste concorrente de filtros (também a capacitores chaveados) construídos através do cascadeamento de filtros biquadráticos (*biquads*). O método consiste em adicionar ao circuito um *biquad* programável, que, durante o

teste, é programado para exercer a mesma funcionalidade do filtro sob teste. Através de um esquema de multiplexação todos os *biquads* do circuito podem ser comparados individualmente ao *biquad* programável. Assim ao invés de duplicar cada bloco do circuito para realizar o teste concorrente, o *biquad* programável é utilizado como bloco redundante para testar todos os outros *biquads* do circuito, reduzindo drasticamente o acréscimo de área. O mesmo esquema de votador (Figura 2.3) é utilizado para determinar se existe ou não diferença entre os sinais de saída do filtro sob teste e do filtro programável.

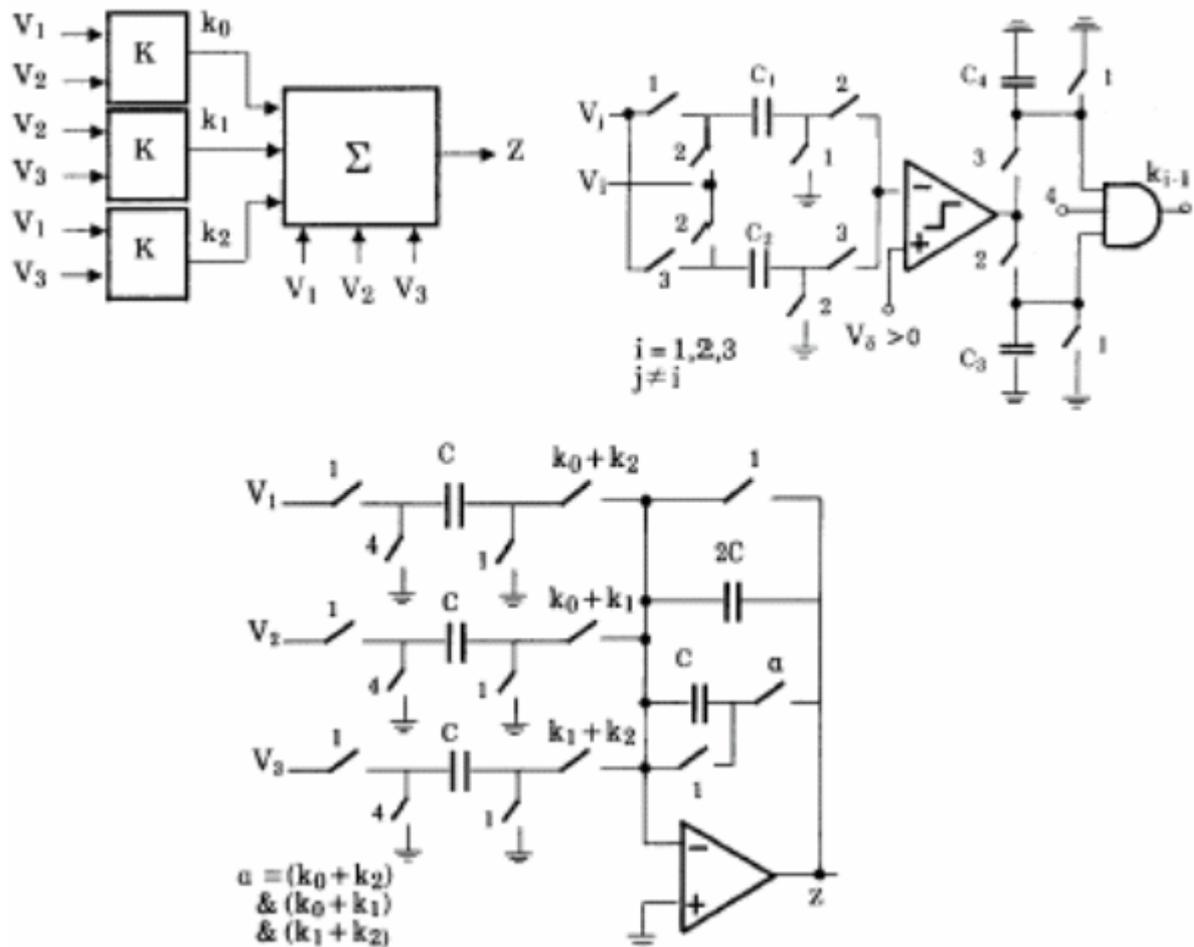


Figura 2.3: Votador analógico e esquemático dos blocos K e  $\Sigma$ . Extraído de ( VÁSQUEZ; RUEDA; HUERTAS, 1991)

Nos trabalhos citados acima o votador pode ser considerado um *checker*, termo que define um bloco operacional cuja função é justamente checar a presença ou ausência de falhas no circuito. Em (KOLARIK *et al.*, 1995) e (KOLARIK; LUBASZEWSKI; CURTOIS, 1993) são propostos *checkers* que possuem tolerância absoluta e relativa (dependendo da amplitude

dos sinais), pois, conforme citado anteriormente, os sinais analógicos podem variar dentro de uma banda de tolerância sem prejudicar a funcionalidade do circuito. Fazendo uma analogia com o teste digital é possível dizer que um *checker* tem a função detectar se um sinal (ou um par de sinais) está dentro ou fora do “código” válido. No caso do teste concorrente, baseado em duplicação, o código é válido quando a diferença dos sinais é menor do que a tolerância considerada, e no caso de circuitos diferenciais o código é válido quando a soma dos sinais nos dois ramos diferenciais considerados está dentro da faixa de tolerância (KOLARIK *et al.*, 1995).

Circuitos totalmente diferenciais são comumente utilizados no projeto analógico por apresentarem características desejáveis quanto a robustez e compensação de efeitos indesejados. O teste de circuitos diferenciais utilizando *checkers* foi abordado em (LUBASZEWSKI *et al.*, 2000). Neste trabalho o *checker* é utilizado para monitorar as entradas de amplificadores diferenciais. Segundo os autores se uma falha afetar algum componente de um dos ramos (positivo ou negativo) do circuito haverá um desbalanceamento nas suas entradas que será detectado pelo *checker*. O trabalho também propõe uma estratégia de teste *off-line* para o *checker*, uma vez que este também pode conter falhas e seu correto funcionamento é crucial para a confiabilidade do teste. Em (VELASCO-MEDINA; NICOLAIDIS; LUBASZEWSKI, 1998), foi proposto um esquema baseado na modificação do projeto dos amplificadores operacionais do circuito para que este apresente uma saída adicional que, associada a um *checker*, facilita a observação de erros durante a operação do circuito.

Em (NEGREIROS; CARRO; SUSIN, 2002) um método de teste analógico *on-line* de baixo custo é apresentado. Tal método é baseado em um amostrador estatístico (*statistical sampler*) que consiste em um circuito simples que pode ser considerado um “conversor A/D de 1 BIT”, pois converte um sinal analógico em um *bitstream* que preserva características estatísticas do sinal como a auto-correlação. Este *bitstream* é então processado utilizando-se a transformada rápida de Fourier (FFT) para estimar a densidade espectral de potência (PSD: *Power Spectrum Density*) da qual pode-se extrair características importantes do sinal. Em 2004 os mesmos autores propuseram a metodologia citada acima como uma alternativa de baixo custo para o teste de circuitos de RF (Rádio Freqüência) (NEGREIROS; CARRO; SUSIN, 2004).

## 2.6 O PADRÃO IEEE 1149.4

Em 1990 um padrão para teste de circuitos digitais foi aprovado pelo IEEE (*Institute of Electrical and Electronic Engineers*) denominado IEEE 1149.1 (IEEE STANDARD ASSOCIATIONS, 2000). Este padrão surgiu da necessidade de automação e melhorias na eficiência do teste de circuitos digitais e consiste na inserção, em nível de projeto, de células de periferia capazes de prover acesso a pontos internos do circuito através de uma cadeia serial de registradores. Os registradores de periferia são dispostos em uma configuração conhecida como *shift register* (registrador de deslocamento). Além de prover um caminho para injeção de vetores de teste e observação de padrões de resposta, os registradores podem ser usados para capturar e injetar sinais nos pinos externos do CI, o que pode ser útil no teste de interconexões de placas de circuito impresso.

Quase uma década mais tarde, em 1999, foi aprovada uma extensão do padrão 1149.1 para circuitos analógicos e de sinal misto: o padrão IEEE 1149.4 (IEEE STANDARD ASSOCIATIONS, 2000). O objetivo é o mesmo do padrão 1149.1: prover acesso a nós internos do circuito através de um barramento e estruturas dedicadas.

No padrão 1149.4 cada pino do CI possui um módulo de periferia. Se este pino for associado a uma entrada ou saída analógica o módulo em questão é o ABM (*Analog Boundary Module*), conforme a Figura 2.4. No caso de pinos digitais o módulo DBM (*Digital Boundary Module*) é idêntico ao utilizado no padrão 1149.1. Cada ABM consiste em um bloco de interconexão disposto entre o núcleo do circuito e o pino, permitindo que o pino seja isolado do núcleo ou conectado a linhas internas do barramento de teste, conforme a Figura 2.5 (OSSEIRAN, 1999).

A estrutura de teste também compreende uma lógica de controle e decodificação de instruções, um bloco de interfaceamento do barramento de teste analógico com os pinos externos (TBIC: *Test Bus Interface Circuit*) e uma porta de acesso ao teste (ATAP: *Analog Test Access Port*). A porta ATAP possui dois pinos a mais do que a porta TAP (*Test Access Port*) utilizada no *boundary scan* digital (IEEE STANDARD ASSOCIATIONS, 1990). Estes pinos (AT<sub>1</sub> e AT<sub>2</sub>) são os pinos de acesso ao barramento de teste analógico. Os outros quatro pinos são os mesmos do padrão digital: TMS (*Test Mode Select*), TCK (*Test Clock*), TDI (*Test Data Input*) e TDO (*Test Data Output*), conforme pode ser visualizado na Figura 2.4.

O padrão IEEE 1149.4 não obteve o mesmo sucesso do seu dual digital no que diz respeito à utilização em grande escala, deixando dúvidas quanto ao seu futuro. Entretanto é

possível encontrar alguns trabalhos relacionados ao 1149.4 propondo maneiras de utilizar e melhorar a performance do teste utilizando este padrão (KAC *et al.* 2003; CALVANO *et al.*, 2002).

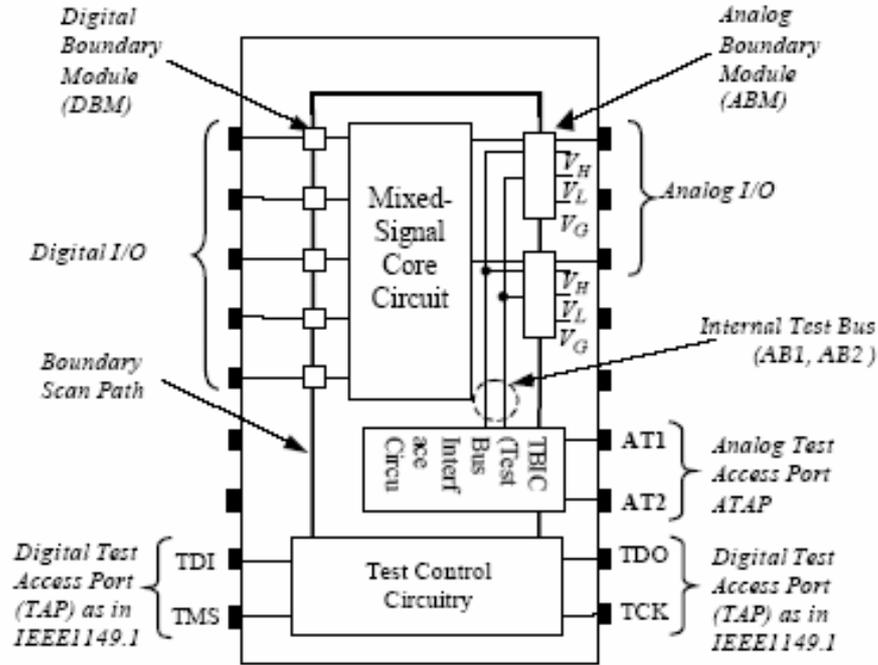


Figura 2.4: Estrutura global de teste compatível com o padrão IEEE 1149.4. Extraída de (OSSEIRAN, 1999)

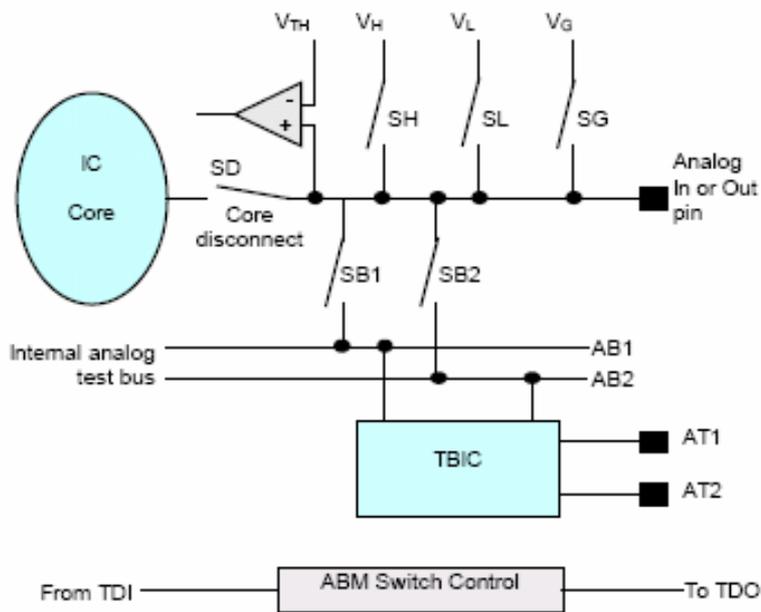


Figura 2.5: Arquitetura de um ABM. Extraída de (OSSEIRAN, 1999)

## 2.7 ANALOG BUILT-IN BLOCK OBSERVER (ABILBO)

Uma estrutura para uso em estratégias de BIST digital capaz de ler e compactar respostas e gerar vetores de teste foi proposta em (KONEMANN; MUCHA; ZWIEHOFF, 1979). Tal estrutura foi denominada BILBO (*Built-In Logic Block Observer*). Uma versão analógica desta estrutura, chamada ABILBO (*Analog Built-In Block Observer*) foi proposta em (LUBASZEWSKI; MIR; PULZ, 1996). Esta estrutura, utilizada no teste analógico, possui as mesmas características de sua versão digital, tais como a capacidade de gerar vetores e de avaliar e compactar respostas de teste. Basicamente, a estrutura do ABILBO é composta por dois integradores e alguns elementos adicionais. No modo de geração de vetores de teste o ABILBO pode ser programado como um oscilador. Já no modo de avaliação de respostas de teste tal estrutura é programada para operar como um duplo integrador (Figura 2.6).

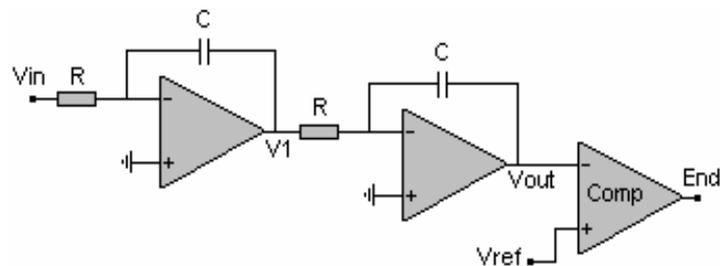


Figura 2.6: Analisador de resposta baseado em duplo integrador

Uma assinatura de um dado sinal pode ser obtida computando-se o tempo decorrido desde a aplicação do sinal na entrada do analisador ( $V_{in}$ ) até que a saída ( $End$ ) atinja uma determinada tensão de referência ( $V_{ref}$ ). Este tempo depende da frequência, da amplitude e do nível DC do sinal.

Assumindo um sinal tal que  $V_{in} = -V_o \sin(\omega t + \phi)$  e  $V_C(t=0) = 0$  para os capacitores do duplo integrador obtém-se as seguintes expressões (LUBASZEWSKI; MIR; PULZ, 1996):

$$V_1 = \frac{V_o}{(\omega\tau)} [\cos\phi - \cos(\omega t + \phi)] \quad (1)$$

$$V_{out} = \frac{V_o}{(\omega\tau)^2} [-\sin(\omega t + \phi) + \omega t \cdot \cos\phi + \sin\phi] \quad (2)$$

Onde  $V_1$  é o sinal na saída do primeiro integrador,  $V_{out}$  é o sinal na saída do segundo integrador e  $\tau = RC$  é a constante de tempo dos integradores.

A Figura 2.7 mostra o efeito da dupla integração sobre um sinal com  $V_o=1V$  e  $f=750\text{ Hz}$ . Considerando  $\tau=0,66\text{ms}$  e  $V_{ref}=2V$  assinatura de tempo obtida é  $t=4,5\text{ms}$ .

Análises sobre o espaço válido dos parâmetros do sinal de entrada (amplitude e frequência) considerando a probabilidade de *aliasing* e a repetibilidade das assinaturas, foram feitas em (LUBASZEWSKI; MIR; PULZ, 1996) e (PEREIRA *et al.*, 2005).

O ABILBO como analisador de resposta foi utilizado em uma das estratégias de teste desenvolvidas neste trabalho e descrita no capítulo 4.

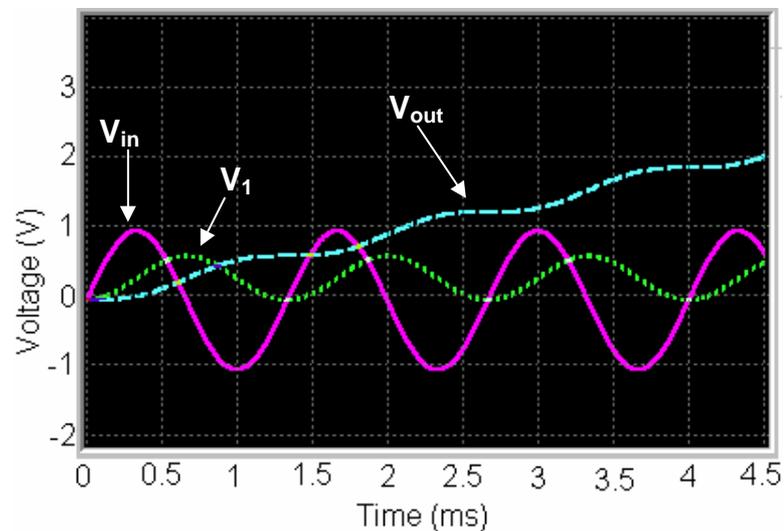


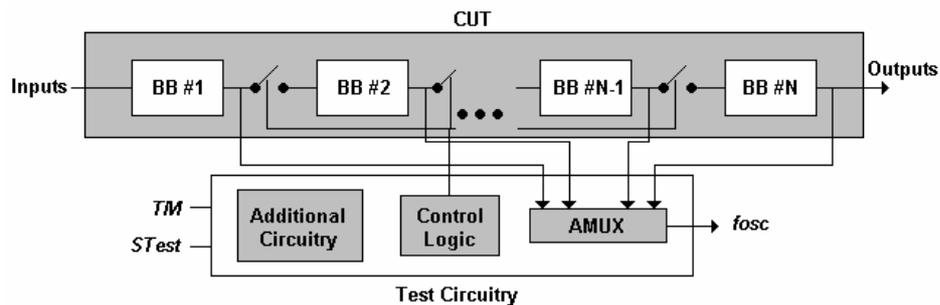
Figura 2.7: Efeito da dupla integração sobre um sinal senoidal

## 2.8 TESTE BASEADO EM OSCILAÇÃO

Entre os anos de 1995 e 1996 ARABI e KAMINSKA propuseram uma metodologia de teste que posteriormente viria a ser amplamente utilizada pela indústria e explorada pela comunidade científica: o teste baseado em oscilação ou OBT (*Oscillation-based Test*) (ARABI; KAMINSKA, 1996). Nesta técnica, também conhecida como OTS (*Oscillation Test Strategy*) ou OBIST (*Oscillation Built-In Self Test*), o circuito sob teste é dividido em blocos funcionais que são convertidos em osciladores. A frequência do sinal gerado por cada bloco oscilador é fixa, e pode ser dita como função dos elementos do CUT. Uma falha que ocorra em um componente do CUT (desvio no valor de um capacitor, por exemplo) pode fazer variar a frequência de oscilação, desde que tal parâmetro seja sensível a variações neste componente. Os sinais de saída destes osciladores são então avaliados e comparados com os

sinais gerados para os circuitos sem falhas (*fault-free*). A presença de uma falha no CUT é identificada quando a frequência e/ou a amplitude (HUERTAS *et al.*, 2002) do sinal gerado pelo bloco, difere dos parâmetros do circuito *fault-free*.

A Figura 2.8 mostra a estrutura de um circuito genérico, onde os blocos funcionais ( $BB\#n$ ), são isolados entre si e convertidos em osciladores durante o teste. Os sinais analógicos gerados por cada um destes osciladores são multiplexados no tempo e avaliados separadamente.



**Figura 2.8:** Estrutura simplificada do teste baseado em oscilação. Adaptada de (ARABI; KAMINSKA, 1996)

Para que uma falha seja detectada esta deve causar uma variação perceptível na frequência ou amplitude de oscilação, ou seja, os parâmetros do sinal devem ser sensíveis a variações no componente falho. Para fins de avaliação do teste define-se como “observabilidade” de uma falha em um determinado componente a sensibilidade da frequência de oscilação em relação a este componente (ARABI; KAMINSKA, 1999).

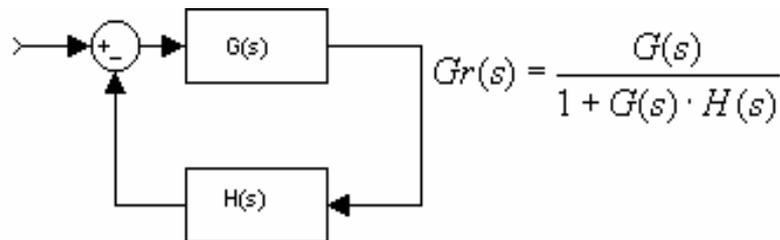
A frequência de oscilação pode variar dentro de uma pequena faixa, devido à tolerância dos componentes e fatores externos como a temperatura e imprecisões no analisador de respostas. Então, falhas que causam uma variação da frequência de oscilação dentro desta faixa, não são detectadas. Por isso, define-se como “banda não detectável de falhas paramétricas” a faixa de variação de um componente que mantém os parâmetros do sinal oscilatório dentro de sua banda de tolerância (ARABI; KAMINSKA, 1999). A determinação da faixa de tolerância para a frequência e amplitude de oscilação pode ser determinada através de análises de “Monte Carlo”, por exemplo.

Existem diversas maneiras de se construir um oscilador. No caso de circuitos de ordem maior ou igual a dois (possuem dois ou mais pólos em sua função de transferência) pode-se adicionar um laço de realimentação. Assim, o circuito pode tornar-se instável e oscilar. Uma

das maneiras de determinar se o circuito oscilará ou não, na presença de uma realimentação, é através do critério de Barkhausen (HAYKIN; VAN VEEN, 2001).

Para um dado sistema com função de transferência em malha aberta  $G(s)$ , com realimentação negativa, cuja rede de realimentação tem função de transferência  $H(s)$  (veja Figura 2.9), a função de transferência do sistema realimentado  $Gr(s)$  será:

$$Gr(s) = \frac{G(s)}{1 + G(s) \cdot H(s)} \quad (3)$$



**Figura 2.9: Representação em blocos de um sistema realimentado e sua função de transferência**

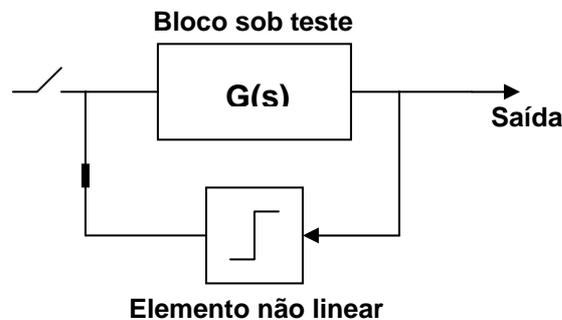
Segundo o critério de Barkhausen, o limiar da instabilidade ocorre quando:

$$H(j\omega) \cdot G(j\omega) = -1 + j0 \quad (4)$$

Em outras palavras, se existir uma frequência para a qual o termo  $G(j\omega)H(j\omega)$ , é unitário, e a rotação total de fase na malha de realimentação for  $2n\pi$  radianos (onde  $n$  é um inteiro), o sistema será instável.

Alguns trabalhos propõem a adição de um elemento não linear no laço de realimentação para garantir o início e a sustentabilidade da oscilação (HUERTAS *et al.* 2002b; VÁZQUEZ *et al.*, 2002). Tal elemento pode ser implementado através de um comparador, conforme a Figura 2.10. Desta maneira, uma oscilação, mesmo que de pequena amplitude, na saída do circuito fará com que um sinal retangular de amplitude controlada seja aplicado na entrada do circuito que se deseja transformar em oscilador. Quando este circuito é originalmente um filtro (passa-baixa ou passa-banda) o sinal retangular na saída do bloco não linear é filtrado, eliminando os componentes harmônicos de alta ordem. O resultado é um sinal “quase senoidal” que será tão mais próximo de um sinal senoidal puro quanto maior for o fator de qualidade do filtro.

Desde a sua proposta o método baseado em oscilação vem sendo utilizado com sucesso no teste de filtros ativos contínuos (ARABI; KAMINSKA, 1999) e a capacitores chaveados (HUERTAS *et al.* 2002b), amplificadores operacionais (ARABI; KAMINSKA, 1998), conversores de dados (ARABI; KAMINSKA, 1997b) e filtros digitais (PERETTI *et al.*, 2005). Além de contemplar o teste de circuitos analógicos e de sinal misto o teste baseado em oscilação pode ser aplicado a diferentes tipos de circuitos puramente digitais. Em 1998 ARABI *et al.* reportaram a aplicação do teste baseado em oscilação a um somador digital de 8 BITS. No teste de circuitos digitais utilizando-se o OBT os osciladores são construídos sob a forma de anel (*ring oscillators*) interconectando-se um número ímpar de inversores. No referido trabalho a aplicação desta técnica visava detectar falhas de atraso (*delay faults*) e falhas de colagem (*stuck at faults*) (ARABI *et al.*, 1998).



**Figura 2.10:** Transformação do bloco sob teste em um oscilador utilizando um elemento de realimentação não linear

Uma importante contribuição ao método de teste baseado em oscilação foi feita por VÁZQUEZ *et al.* em 2002. No trabalho em questão é proposto um esquema de baixo custo (segundo os autores) para a avaliação “intrachip” dos parâmetros de interesse do sinal oscilatório. O analisador da resposta oscilatória consiste em um modulador sigma-delta que traduz o sinal gerado pelo oscilador em um *bitstream*. Este *bitstream* é então processado digitalmente obtendo-se os valores da amplitude, frequência e nível DC do sinal. A Figura 2.11 exemplifica tal esquema.

A precisão deste método depende principalmente da taxa de sobre amostragem (OSR: *Over Sample Ratio*) do modulador sigma-delta. Uma maneira de se reduzir o acréscimo de área é realizar a extração dos parâmetros (frequência, amplitude e nível DC) com auxílio de testadores externos. Assim apenas a área do modulador influenciaria negativamente na área ocupada em silício e o teste poderia ser realizado por um ATE puramente digital (VÁZQUEZ *et al.*, 2002).

Um problema inerente ao teste baseado em oscilação é a garantia do início (*start up*) e sustentabilidade da oscilação. VÁZQUEZ *et al.*, em 2002, propuseram modificações estruturais aos amplificadores operacionais para que estes apresentem características que garantam um rápido *start up* e sustentabilidade da oscilação quando utilizados na aplicação do OBT (VÁZQUEZ *et al.*, 2002b).

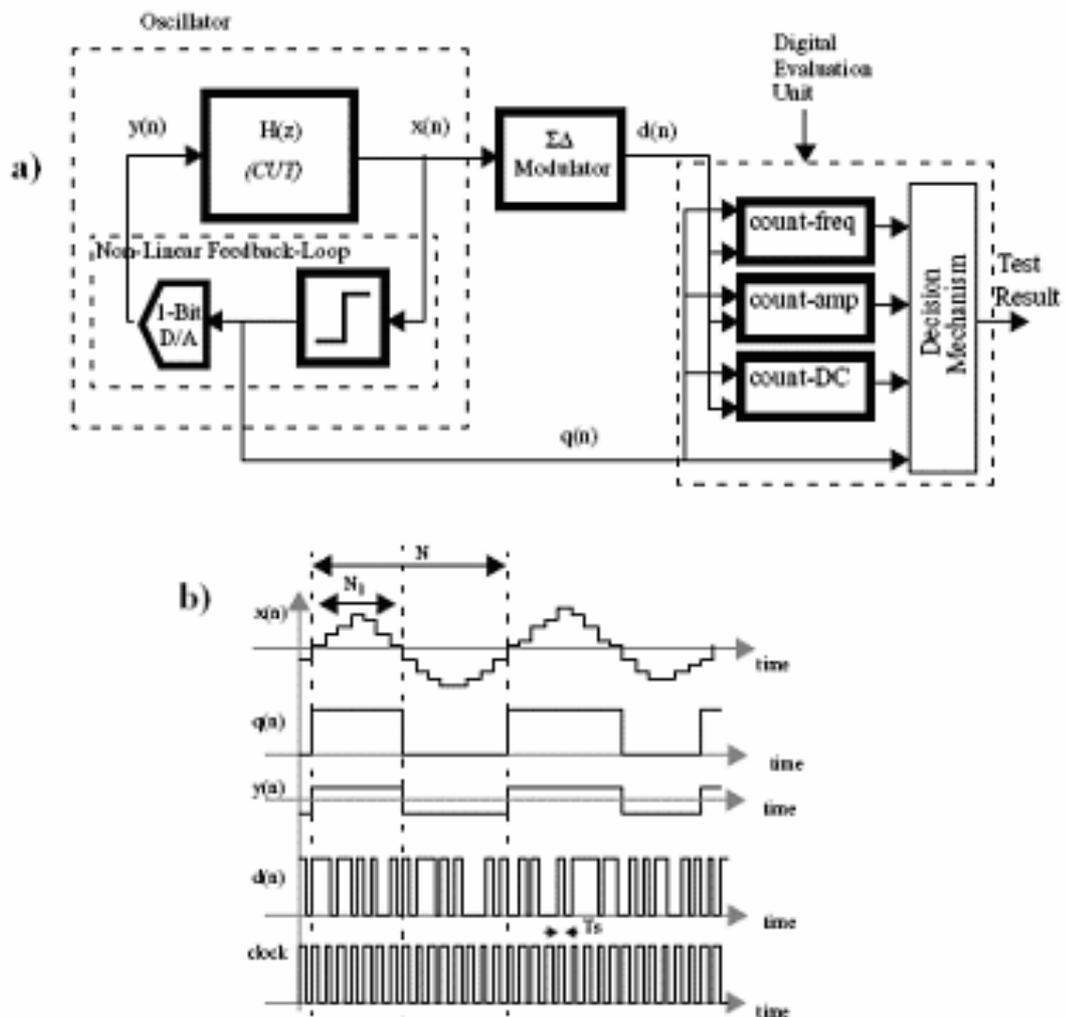


Figura 2.11: (a) Esquema de avaliação intrachip. (b) Formas de onda. Extraído de (VÁZQUEZ *et al.*, 2002)

## 2.9 MÉTODO DE TESTE BASEADO NA ANÁLISE DE RESPOSTA TRANSIENTE

O método de teste baseado em análise de resposta transiente (TRAM: *Transient Response Analysis Method*) (CALVANO; ALVES; LUBASZEWSKI, 1999, 2000) foi proposto como uma alternativa para o teste de filtros e outras funções analógicas. Neste método o circuito sob teste é dividido em blocos de primeira e segunda ordem. Durante o

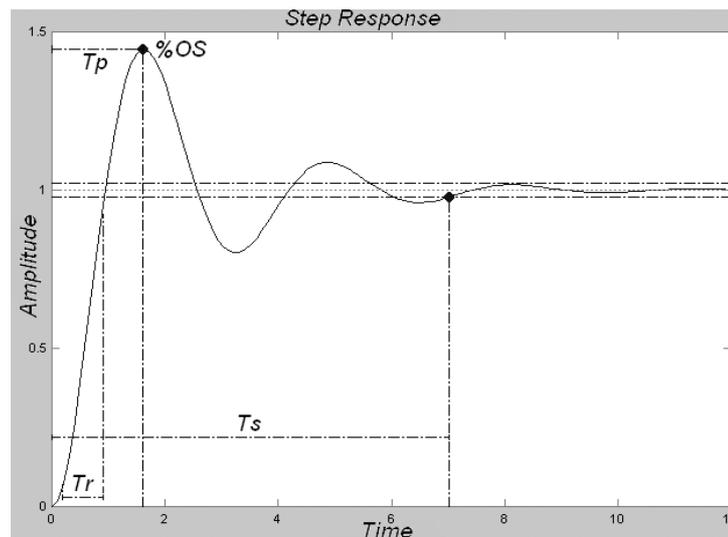
teste um estímulo como um degrau (*step*), rampa ou parábola, dependendo da função do bloco a ser testado, é aplicado ao bloco alvo e o regime transiente das resposta é analisado.

Considerando o bloco sob teste como um filtro passa-baixa de segunda ordem cuja função de transferência é dada por:

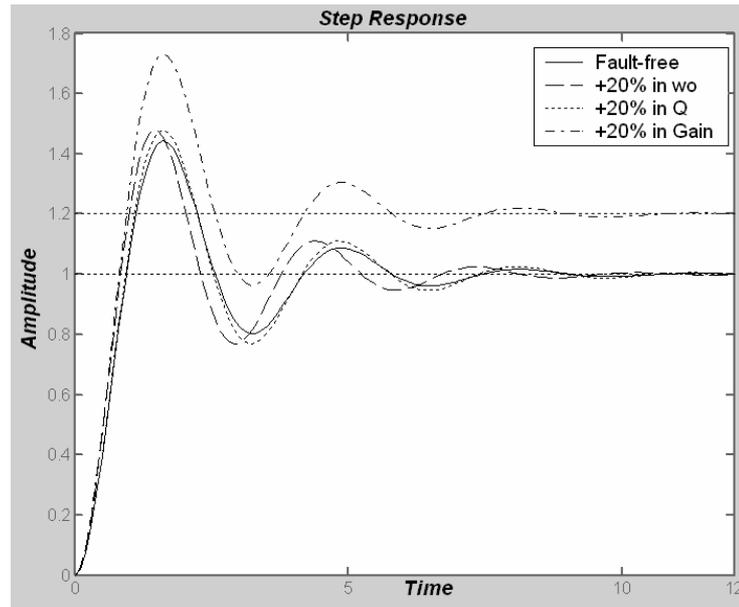
$$H_{LP}(s) = \frac{\omega_0^2 \cdot G}{s^2 + \left(\frac{\omega_0}{Q}\right) \cdot s + \omega_0^2} \quad (5)$$

Se houver uma falha em algum parâmetro do bloco como a frequência de corte ( $\omega_0$ ), fator de qualidade ( $Q$ ) e ganho na faixa de passagem ( $G$ ), um ou mais parâmetros da resposta transiente a um estímulo de entrada do tipo *step* irá(ao) diferir do(s) parâmetro(s) da resposta nominal (sem falhas) (CALVANO; ALVES; LUBASZEWSKI, 1999). Estes parâmetros são o tempo de subida ( $Tr$ ), tempo de pico ( $Tp$ ), pico de sobre-tensão percentual (*Percent Over Shoot* - %OS), tempo de acomodação ( $Ts$ ) e valor final da amplitude no estado estacionário, conforme mostrado na Figura 2.12. A Figura 2.13 mostra as respostas ao *step* de um filtro de segunda ordem com variações de +20% na frequência de corte, fator de qualidade e ganho (falhas injetadas individualmente).

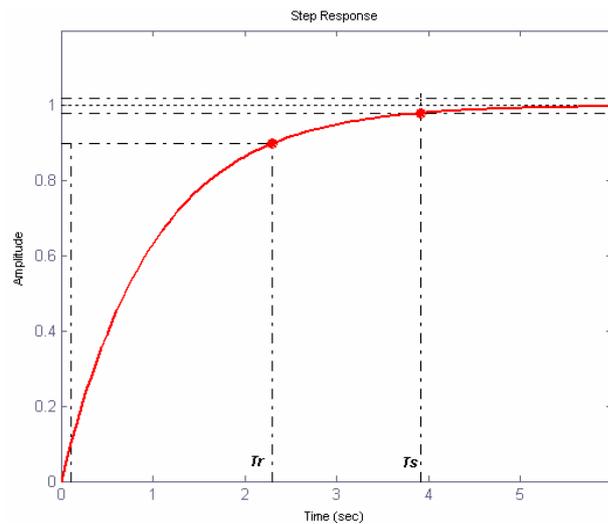
O mesmo se aplica a funções de primeira ordem, porém o número de parâmetros da resposta transiente é reduzido, consistindo no tempo de subida ( $Tr$ ), tempo de acomodação ( $Ts$ ) e valor final da amplitude, conforme pode ser visto na Figura 2.14.



**Figura 2.12: Resposta ao *step* de um filtro passa-baixa de segunda ordem**



**Figura 2.13:** Respostas ao *step* de um filtro passa-baixa de segunda ordem com desvios de +20% nos parâmetros da função de transferência



**Figura 2.14:** Resposta ao *step* de um sistema de primeira ordem

É possível aplicar este método também a filtros passa-alta, passa-faixa e rejeita-faixa, para isso basta fazer uma transformação em frequência (CALVANO; ALVES; LUBASZEWSKI, 1999). Observando as equações das funções de transferência dos filtros listados abaixo percebe-se que estas diferem da função passa-baixa pela ordem do polinômio do numerador. Os índices LP (Low-Pass), HP (High-Pass) e BP (Band-Pass) referem-se ao tipo do filtro (passa-baixa, passa-alta e passa-banda respectivamente).

Primeira ordem:

Passa-baixa:

$$H_{LP1}(s) = \frac{G\omega_o}{s + \omega_o} \quad (6)$$

Passa-alta:

$$H_{HP1}(s) = \frac{Gs}{s + \omega_o} \quad (7)$$

Segunda ordem:

Passa-baixa:

$$H_{LP}(s) = \frac{\omega_o^2 \cdot G}{s^2 + \left(\frac{\omega_o}{Q}\right) \cdot s + \omega_o^2} \quad (8)$$

Passa-alta:

$$H_{HP}(s) = \frac{G \cdot s^2}{s^2 + \left(\frac{\omega_o}{Q}\right) \cdot s + \omega_o^2} \quad (9)$$

Passa-banda:

$$H_{BP}(s) = \frac{\omega_o \frac{G}{Q} \cdot s}{s^2 + \left(\frac{\omega_o}{Q}\right) \cdot s + \omega_o^2} \quad (10)$$

Desta maneira, para realizar a transformação em frequência basta integrar (multiplicar em frequência por  $1/s$ ) a função de transferência uma ou duas vezes, a fim de se eliminar o termo em  $s$  do numerador das funções. Do ponto de vista sistêmico esta integração estará em série com o sistema alvo (filtro), o que equivale a integrar o estímulo aplicado, neste caso um *step*. A Figura 2.15 ilustra este processo considerando um filtro passa-banda. A integração de um *step* gera uma rampa, e a dupla integração do *step* (integração da rampa) gera uma parábola. Desta maneira, para as funções em que a ordem do polinômio do numerador é zero, aplica-se um *step*, se o numerador for de ordem 1, aplica-se uma rampa, e finalmente, se o polinômio do numerador for de segunda ordem, o sinal de entrada deve ser uma parábola.

Assim, independentemente do tipo de filtro (passa-alta, passa-banda ou passa-faixa), as respostas serão análogas às Figuras 2.12 e 2.14 (segunda e primeira ordem, respectivamente) (CALVANO; ALVES; LUBASZEWSKI, 1999).

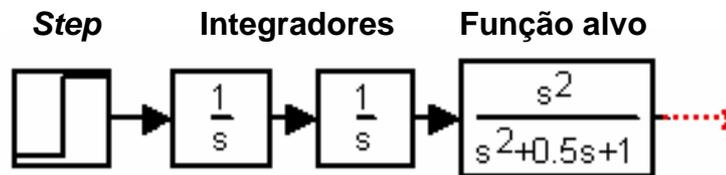


Figura 2.15: Transformação em frequência através da integração do estímulo de entrada

O teste baseado em análise de resposta transiente foi aplicado a um dos FPAAs considerados neste trabalho, conforme pode ser observado com detalhes no capítulo 5.

## 2.10 TESTE DE FPAAS

Os FPAAs são componentes relativamente recentes na indústria eletrônica e no meio acadêmico. Por este motivo, existem poucos trabalhos a respeito do teste deste tipo de dispositivo. Em trabalhos anteriores a técnica de teste baseada em oscilação foi utilizada para testar os blocos analógicos configuráveis de um FPAA da *Lattice<sup>TM</sup> Semiconductors* (LATTICE; 2000) cuja programabilidade é estrutural, ou seja, é possível programar diretamente valores de componentes. Em uma das abordagens a análise dos parâmetros oscilatórios é feita externamente o que configura uma estratégia de BIST parcial (BALEN *et al.*, 2005). Em outra abordagem um analisador de resposta foi construído internamente com os recursos do dispositivo sob teste. Desta maneira um esquema de BIST total foi desenvolvido (BALEN *et al.*, 2004).

As mesmas técnicas utilizadas nos trabalhos acima citados, associadas a um algoritmo para encontrar caminhos críticos na rede de interconexão, foram utilizadas para testar os recursos de interconexão de um FPAA da primeira geração da *Anadigm Company* (ANDRADE JR. *et al.*, 2005). Em um trabalho recente o teste funcional dos blocos analógicos configuráveis de um FPAA do mesmo fabricante foi abordado (BALEN *et al.*, 2006).

O teste *on-line* dos blocos analógicos programáveis de um FPAA da Motorola (atualmente pertencente à *Anadigm*) foi abordado em (WANG; KILKARNI; TRAGOUDAS, 2004). Neste trabalho os circuitos programados nos FPAAs são particionados em blocos simples que são então duplicados e suas saídas são comparadas, visando detectar desvios ocasionados por falhas.

A injeção de falhas em FPAAs modificando-se o *bitstream* de programação do componente foi proposta em um trabalho recente (SLAUGHTER; STROUD, 2003). Em outro trabalho recente (BALEN; LUBASZEWSKI; RENOVELL, 2006) tal método de injeção de falhas foi utilizado para avaliar o impacto de *BIT flips* (inversões) nos BITS de programação de um FPAA o que pode prejudicar o correto funcionamento das funções programadas no dispositivo. Tal evento é conhecido como SEU (*Single Event Upset*) e pode ser ocasionado devido à incidência de radiação ou partículas eletricamente carregadas (MESSENGER, 1992).

Alguns trabalhos acima citados originaram-se da linha de pesquisa desenvolvida neste trabalho de mestrado e serão abordados com detalhes nos capítulos subseqüentes desta dissertação.

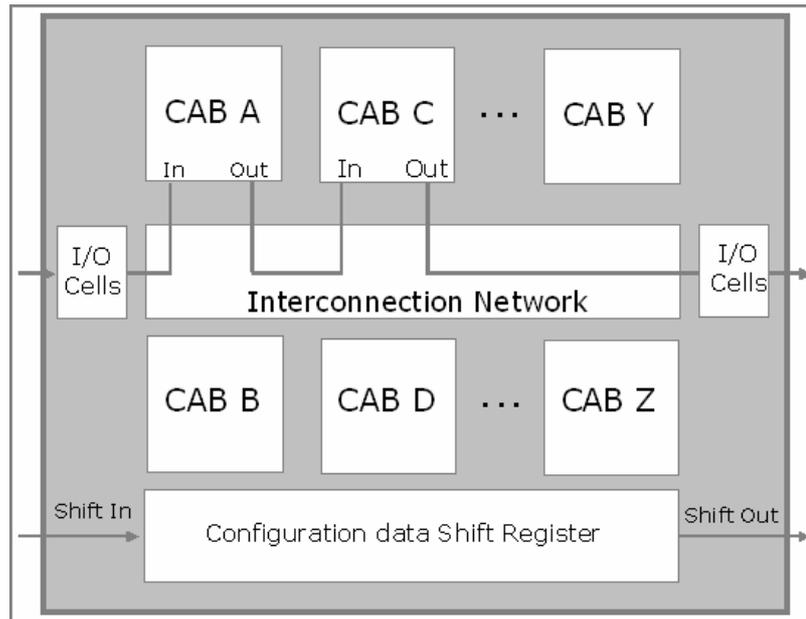
### 3. FIELD PROGRAMMABLE ANALOG ARRAYS (FPAAS)

#### 3.1 ARQUITETURA TÍPICA DE UM FPAA GENÉRICO

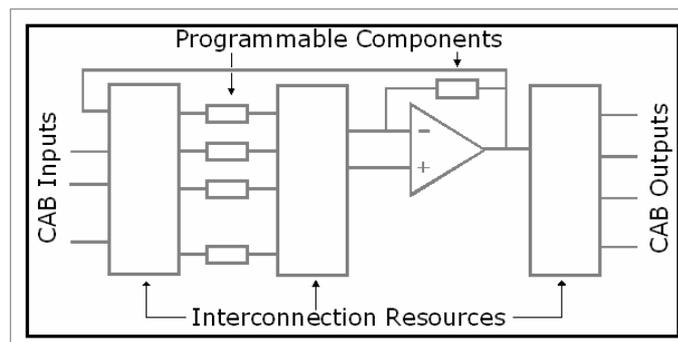
Os FPAAs são circuitos analógicos programáveis que podem ser reconfigurados durante as etapas de desenvolvimento de um projeto bem como em campo, durante a utilização do sistema no qual o componente se encontra (daí o seu nome, que traduzido para o português significa “Arranjo Analógico Programável em Campo”).

O FPAA proporciona ao domínio analógico as mesmas características que o seu dual digital, o FPGA (*Field Programmable Gate Array*), proporciona aos circuitos digitais, aumentando a flexibilidade e diminuindo o tempo de projeto. Outra característica de alguns FPAAs é a reconfiguração dinâmica, o que pode ser uma ótima alternativa em sistemas de controle e instrumentação adaptativos ou hardware evolutivo (HEREFORD; PRUITT, 2004; ZNAMIROWSKI; PAULUSINSKI; VRUDHULA, 2004), onde algumas características do circuito precisam ser alteradas de acordo com as mudanças nos valores das variáveis consideradas nos processos. Devido à disponibilidade de ferramentas de programação automatizadas e de fácil manipulação, o projeto de circuitos analógicos complexos pode ser realizado sem a necessidade de um projetista analógico altamente especializado, tornando o desenvolvimento de sistemas analógicos uma tarefa menos difícil e demorada.

A Figura 3.1 ilustra a arquitetura típica de um FPAA genérico formada por blocos analógicos programáveis (*CABs – Configurable Analog Blocks*), células de entrada e saída (*I/O*), uma rede de interconexões e registradores de memória, onde são armazenados os dados digitais que programam o componente. A rede de interconexões é responsável por conectar os diferentes CABs entre si e também com as células de I/O. As células de I/O são a interface do FPAA com o sistema externo e podem ser compostas por *buffers*, filtros *anti-aliasing* ou *smoothing*, entre outras funções de condicionamento de sinal.



**Figura 3.1: Arquitetura de um FPA genérico**



**Figura 3.2: Arquitetura de um CAB genérico**

A Figura 3.2 ilustra o esquemático genérico de um bloco analógico configurável (CAB) de um FPA. Cada CAB é composto por um conjunto de componentes analógicos programáveis, blocos de interconexões e um amplificador operacional de saída. Os componentes de um CAB podem ser configurados como simples linhas de interconexão (fios), resistores, capacitores e transdutores, por exemplo. Em geral os parâmetros programáveis dos CABs são ganho de amplificadores, valores de resistores e capacitores bem como a habilitação de laços de realimentação locais ou globais.

### 3.2 FPAAS ACADÊMICOS

Pesquisas na área de circuitos analógicos programáveis resultaram em diversas propostas de arquiteturas de FPAAs nos últimos anos. Nesta seção são apresentados alguns FPAAs concebidos ou projetados em universidades ou institutos de pesquisa que não chegaram a ser comercializados.

Em (LEE; GULAK, 1991) um FPAa baseado em transistores CMOS operando na região *sub-threshold* (abaixo da tensão de limiar) foi proposto. A célula configurável deste FPAa é programada por 3 BITS armazenados em um *shift register*. Cada BIT configura o estado de um conjunto de chaves que interconectam elementos primitivos conforme a numeração na Figura 3.3(a). Os elementos primitivos são mostrados na Figura 3.3(b). Neste esquema é possível visualizar as chaves controladas por cada BIT, numeradas como 1, 2 e 3. Estes números representam a posição do BIT no *shift register* de programação (canto inferior direito da Figura 3.3(a)).

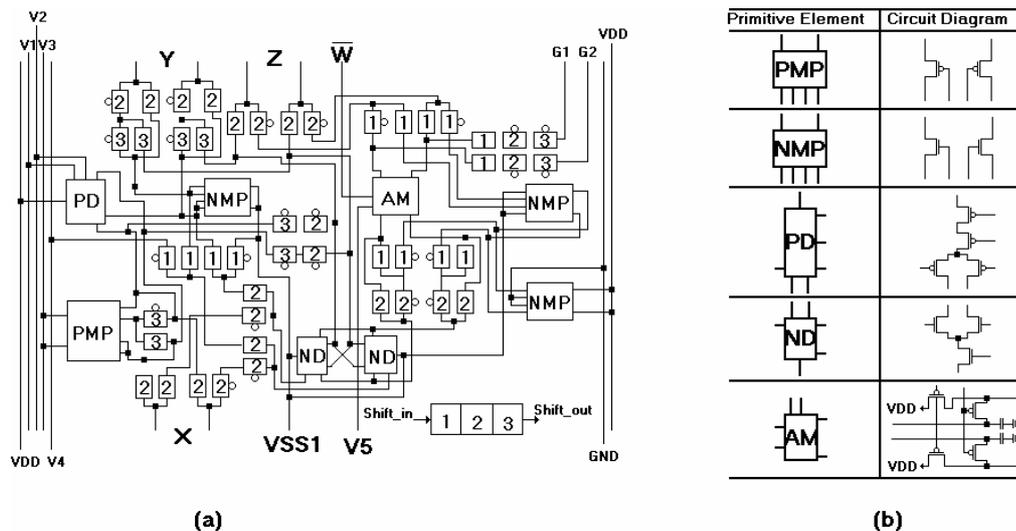


Figura 3.3: (a) Célula configurável e (b) elementos primitivos da arquitetura proposta em (LEE; GULAK, 1991)

Posteriormente os mesmos autores propuseram uma nova arquitetura de FPAa baseada na estrutura acima, porém com transcondutores (ao invés de transistores) realizando a interconexão dos blocos primitivos (LEE; GULAK, 1992). A principal vantagem sobre a versão anterior é o fato dos elementos de interconexão atuarem também como resistores e blocos de ganho programáveis. Uma modificação na topologia dos transcondutores utilizados foi realizada a fim de diminuir a área ocupada pelo FPAa dando origem a uma nova versão do dispositivo (LEE; GULAK, 1995). Esta estrutura consiste em 4 CABs e 8 redes de

interconexões contendo resistores variáveis além de outros recursos de interconexão. Cada CAB é composto por um *OPAMP* com capacitores de realimentação habilitáveis (podem ser conectados ou desconectados do *OPAMP* por meio de chaves programáveis) em uma configuração totalmente diferencial, conforme pode ser visualizado na Figura 3.4.

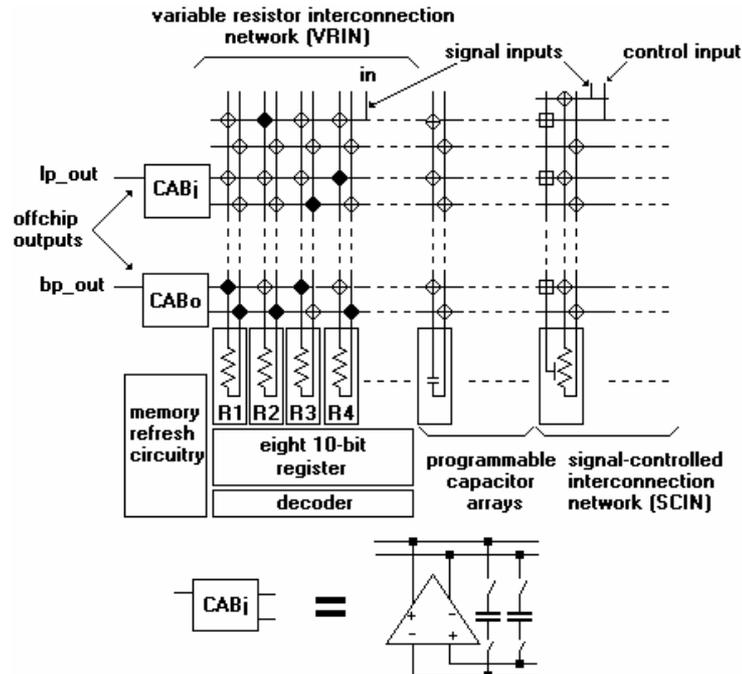


Figura 3.4: Arquitetura do FPAA proposto em (LEE; GULAK, 1995)

Em (LEE; HUI, 1998) um FPAA a capacitor chaveado também baseado em estruturas totalmente diferenciais é apresentado. A arquitetura proposta é composta por um conjunto de CABs em cuja vizinhança há dois bancos de capacitores programáveis, conforme a Figura 3.5. Uma rede de interconexão permite que os CABs e capacitores de cada linha sejam conectados aos CABs e capacitores de outras linhas na matriz configurável. A estrutura do CAB e do banco de capacitores são descritas na Figura 3.6.

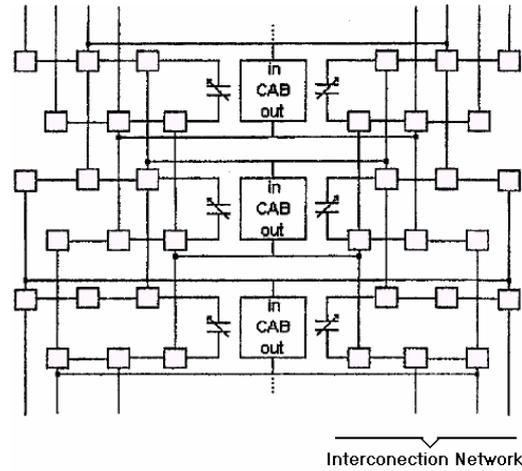


Figura 3.5: Arquitetura do FPA proposto em (LEE; HUI, 1998)

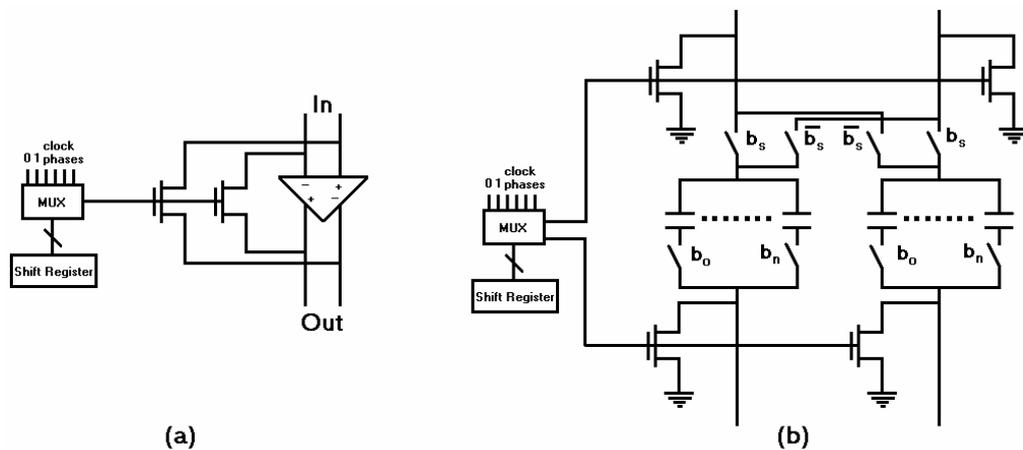


Figura 3.6: (a) Arquitetura do CAB e (b) do banco de capacitores do FPA proposto em (LEE; HUI, 1998)

Em (KUTUK; KANG, 1998) um FPA a capacitores chaveados é apresentado. O CAB é composto por dois amplificadores operacionais cujas configurações adicionais permitem a implementação de dois integradores, um puro e um com perdas (Figura 3.7). Chama-se integrador puro o circuito que implementa a Função de transferência mostrada na Equação 11, composta por um único pólo na origem. Já o integrador com perdas é um filtro passa-baixas (Equação 6), que apresenta um único pólo em uma frequência diferente de zero. Tal estrutura é ideal para implementação de *biquads*. Adicionalmente, os integradores podem ser desconectados mutuamente formando dois blocos distintos. A interconexão dos CABs é feita pela rede mostrada na Figura 3.8, a qual possui um conjunto de capacitores chaveados que podem ser utilizados como elementos programáveis adicionais.

A frequência máxima de relógio que se pode utilizar neste FPAA é 256 KHz, logo, segundo o teorema da amostragem de Nyquist a banda de operação do dispositivo é limitada em 128 KHz.

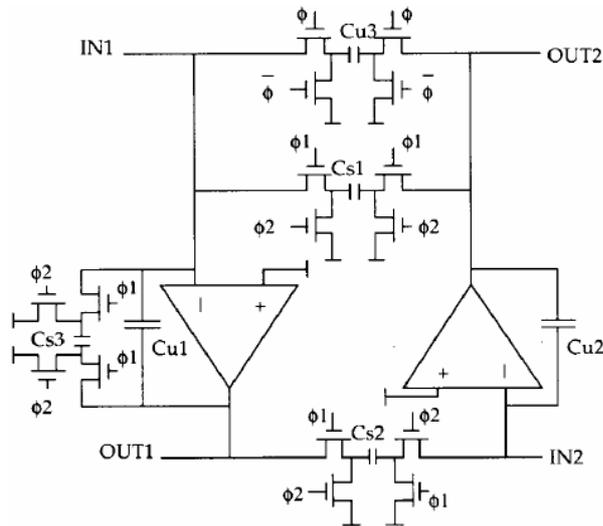


Figura 3.7: CAB do FPAA proposto em (KUTUK; KANG, 1998)

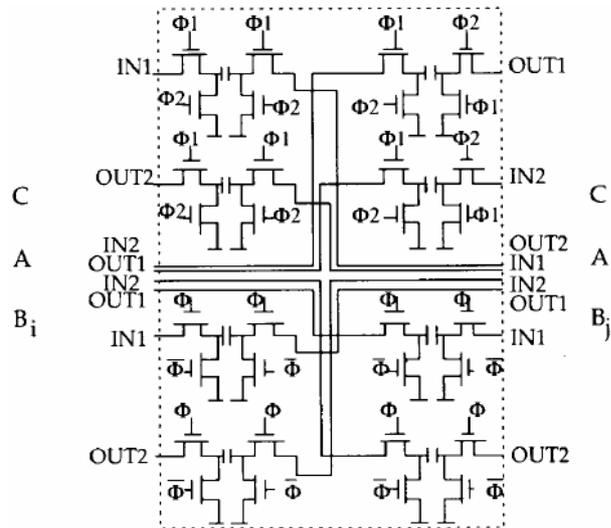


Figura 3.8: Bloco de interconexão do FPAA proposto em (KUTUK; KANG, 1998)

Em (PREMONT *et al.* 1998) é apresentado um FPAA cuja estrutura é baseada em um bloco analógico conhecido como *current conveyor* (carregador de corrente). A Figura 3.9 mostra o esquema e a representação do carregador de corrente.

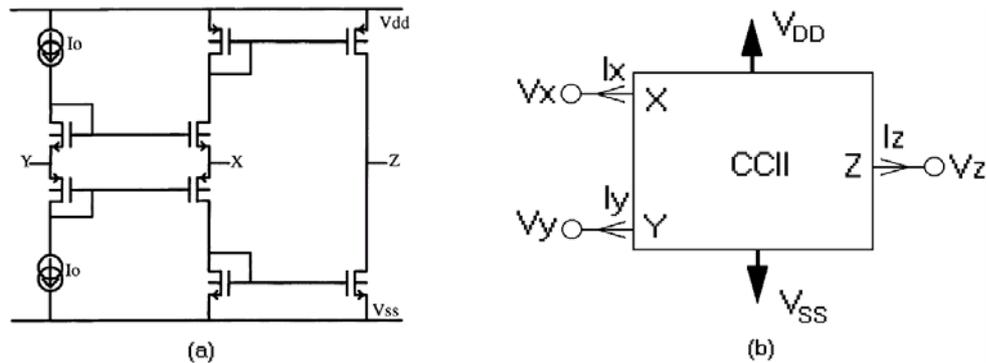


Figura 3.9: (a) Esquemático e (b) simbologia do carregador de corrente segundo (PREMONT, 1998)

Considerando o esquema da Figura 3.9(b) têm-se que uma tensão aplicada ao terminal X produzirá a mesma tensão no terminal Y. De maneira análoga, uma corrente aplicada no terminal X resultará em uma igual corrente fluindo no terminal Y e a mesma quantidade de corrente será “carregada” para o terminal Z. Assim o terminal Z tem uma característica de fonte de corrente com uma alta impedância de saída (SEDRA; ROBERTS; GOHH, 1990). Com tal estrutura é possível obter-se fontes de tensão e corrente controladas, e com a adição de componentes passivos é possível implementar os seguintes blocos que operam em modo corrente: amplificador, derivador, integrador e somador (SEDRA; ROBERTS; GOHH, 1990). A principal característica do carregador de corrente é que o ganho não depende da faixa de frequência de operação como no caso dos *OPAMPS*. Com esta estrutura conseguiu-se alcançar uma faixa de frequência de operação de 3 MHz (PREMONT *et al.* 1998).

Outros trabalhos que descrevem FPAA's baseado em carregadores de corrente ou que operam em modo corrente foram descritos em (GAUDET; GULAK, 1997; ZHANG; BRATT; MACBETH, 1996).

### 3.3 FPAA'S COMERCIAIS

Das cinco empresas que já comercializaram FPAA's apenas três ainda continuam a produção deste tipo de dispositivo. O primeiro FPAA de tempo contínuo a surgir no mercado (não mais disponível hoje em dia) foi o TRAC (*Totally Reconfigurable Analog Circuit*) (ZETEX, 1999). Este dispositivo consiste em 20 CABs que podem ser interligados localmente (internamente) e de modo global (externamente). O CAB do TRAC permite a ligação de transistores bipolares de maneira a implementar amplificadores logarítmicos. As funções possíveis de se implementar com o TRAC são: soma, inversão, logaritmo, anti-logaritmo, retificação e ganho. A utilização de somadores associados a amplificadores *log* e *anti-log*

permite a implementação de funções matemáticas tais como multiplicação e divisão. Por isso é possível afirmar que este FPPA era especializado em tratamento matemático de sinais analógicos. A Figura 3.10 mostra os blocos analógicos que podem ser implementados neste dispositivo. O diagrama em blocos da estrutura global do FPPA TRAC20LH é mostrado na Figura 3.11.

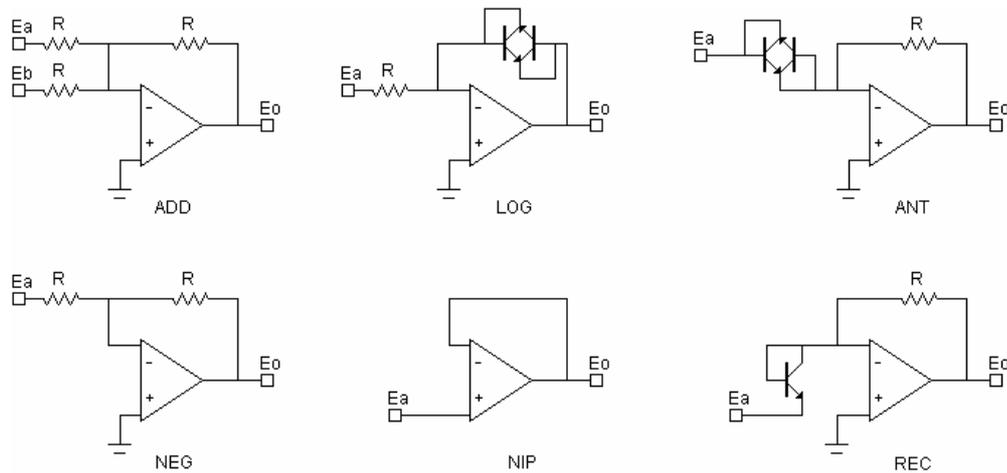


Figura 3.10: Funções analógicas implementáveis no TRAC20LH (ZETEX, 1999)

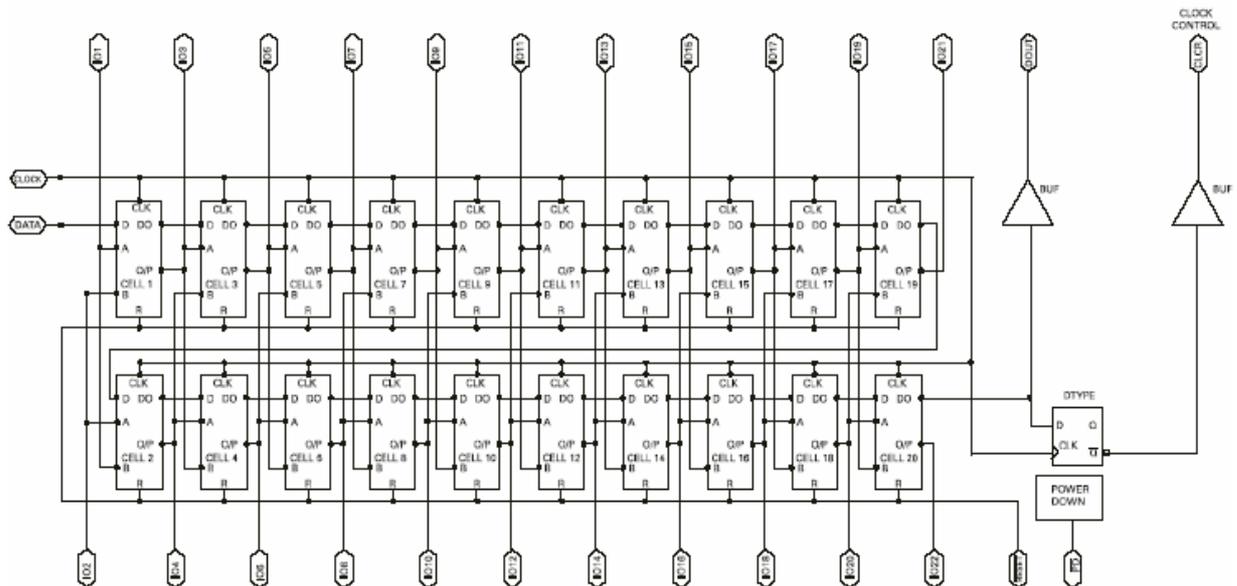


Figura 3.11: Estrutura global do FPPA TRAC20LH da Zetex (ZETEX, 1999)

Outro FPPA cuja produção foi descontinuada fazia parte de um sistema *On-Chip* (SoC), trata-se do FIPSoC (*Field Programmable System-on-Chip*) da Sidsa (SIDSA, 2005). Este dispositivo não será descrito neste trabalho devido à indisponibilidade de informação por parte do fabricante e também por não adicionar informações relevantes ao trabalho.

Ainda no contexto de SoCs é possível encontrar (ainda disponível no mercado) outra família de FPAA's embutidas em sistemas mistos: a família CY8C2XXXX da Cypress, também referenciada como PSoC (*Programmable System-on-Chip*) (CYPRESS, 2002). Os componentes da família PSoC possuem blocos analógicos de tempo contínuo e a capacitores chaveados, além de elementos digitais e mistos, como um microcontrolador, memórias e conversores de dados. A representação em blocos dos componentes da família PSoC é mostrada na Figura 3.12. As Figuras 3.13 e 3.14 ilustram, respectivamente, os blocos analógicos programáveis de tempo contínuo e a capacitores chaveados do PSoC.

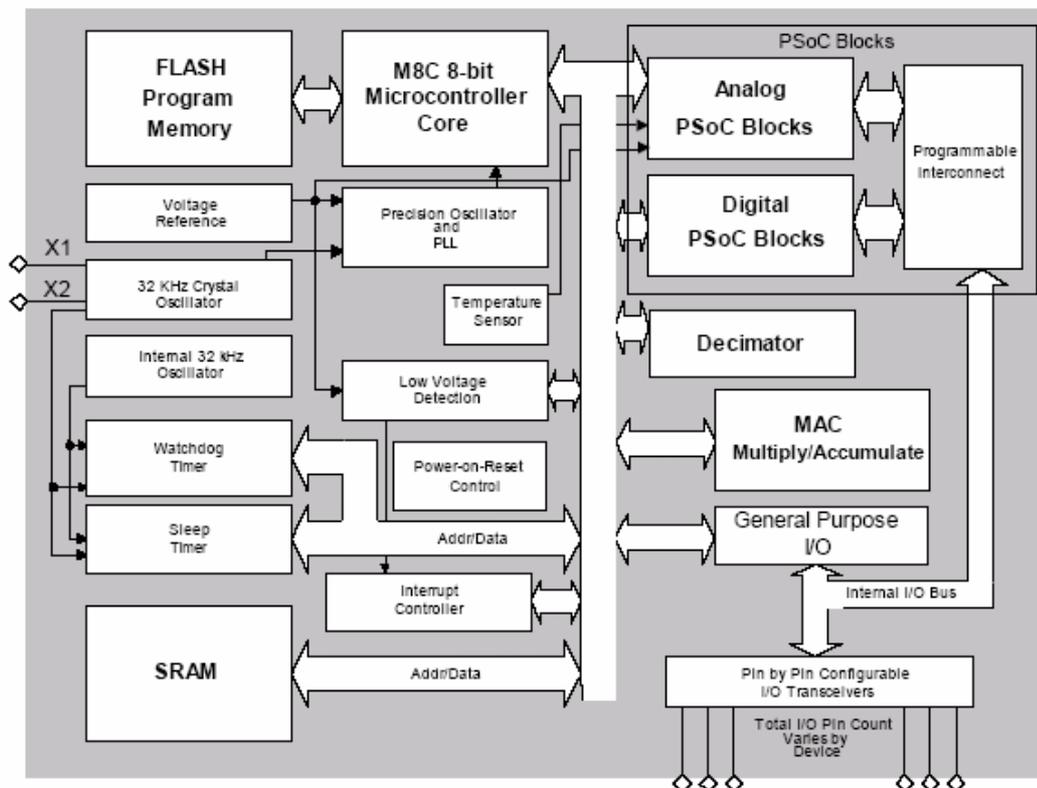


Figura 3.12: Representação em blocos do PSoC (CYPRESS, 2002)

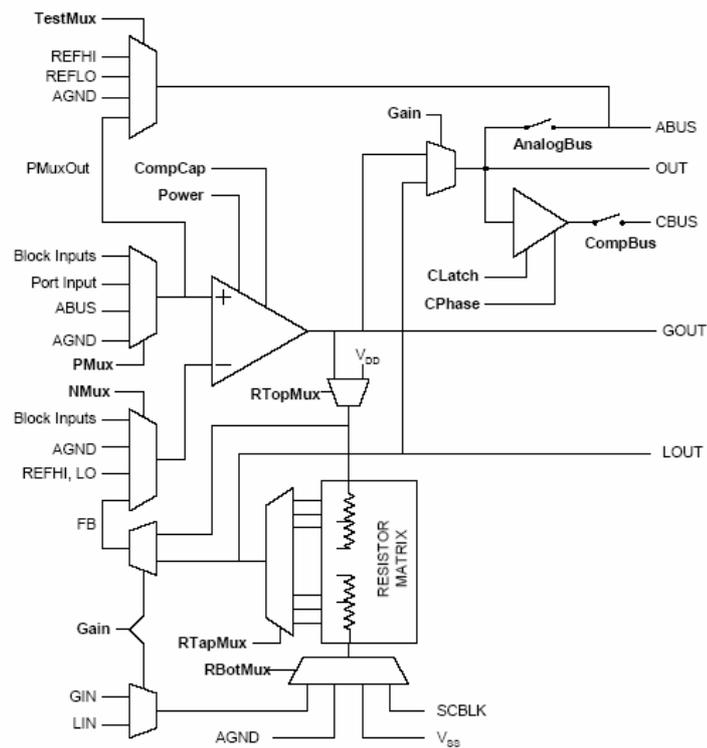


Figura 3.13: Esquemático da célula analógica de tempo contínuo do PSoC (CYPRESS, 2002)

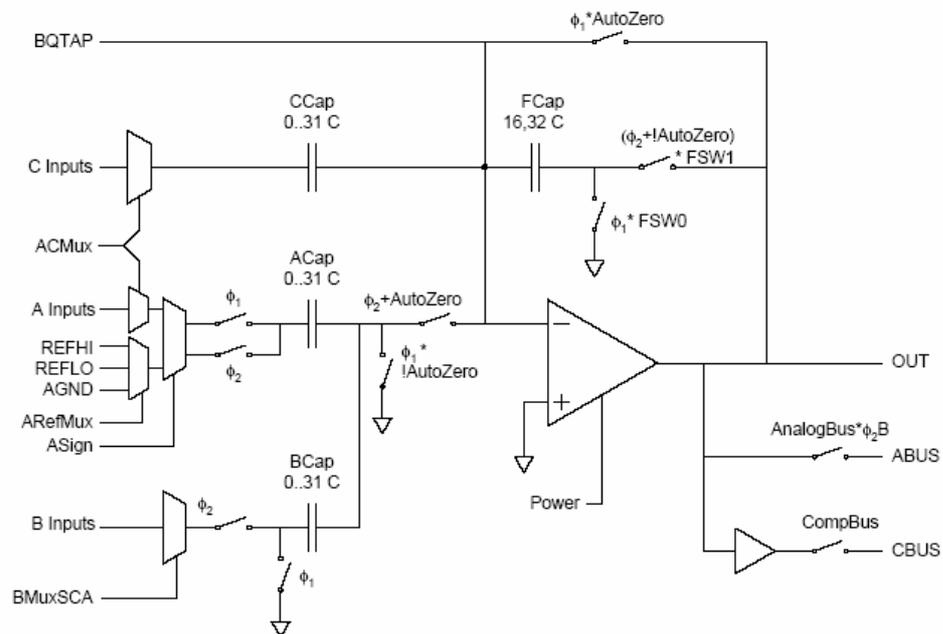


Figura 3.14: Esquemático da célula analógica a capacitores chaveados do PSoC (CYPRESS, 2002)

Os FPAAs acima apresentados não foram explorados nos experimentos práticos deste trabalho. As seções 3.3.1 a 3.3.3 descrevem os FPAAs para os quais as técnicas e estratégias de teste consideradas foram desenvolvidas ou aplicadas.

### 3.3.1 A Família ispPAC da *Lattice Semiconductors*

A família ispPAC (*In-System Programmable Analog Circuits*) teve origem em uma estrutura desenvolvida em 1995 pela IMP (*International Microelectronics Products*) hoje em dia pertencente a *Lattice Semiconductors*. Uma das principais características dos componentes da família ispPAC é a memória de programação do tipo EEPROM (*Electrical Erasable Programmable Read Only Memory*) (LATTICE, 2000).

A família ispPAC possui 6 componentes. Os componentes ispPAC10, ispPAC20, ispPAC30, são FPAA's que possuem uma maior versatilidade em comparação com os outros membros da família ispPAC, podendo realizar funções analógicas como filtros, somadores, integradores, e amplificadores. Já os componentes ispPAC80, ispPAC81 são filtros programáveis de quinta ordem com topologia definida e não alterável. A programabilidade destes dois últimos componentes se restringe aos parâmetros (ganho, frequência de corte e fator de qualidade) e ao tipo dos filtros (Butterworth, Chebychev, Elíptico, entre outros). A diferença entre os componentes ispPAC80 e ispPAC81 é apenas a faixa de operação em frequência. A Figura 3.15 mostra a representação global dos componentes ispPAC10, ispPAC20 e ispPAC30.

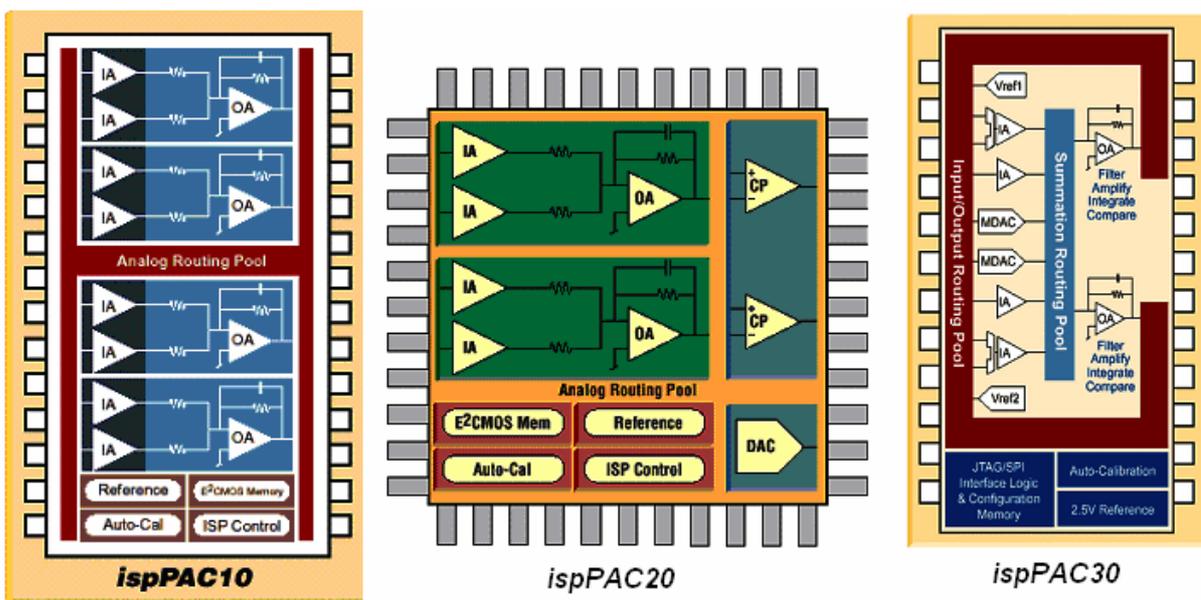
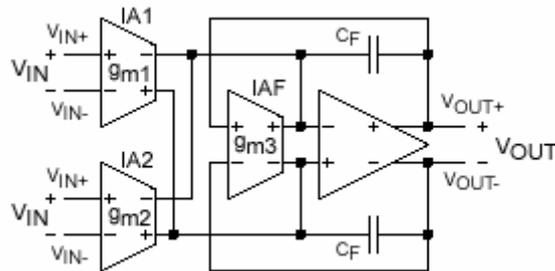


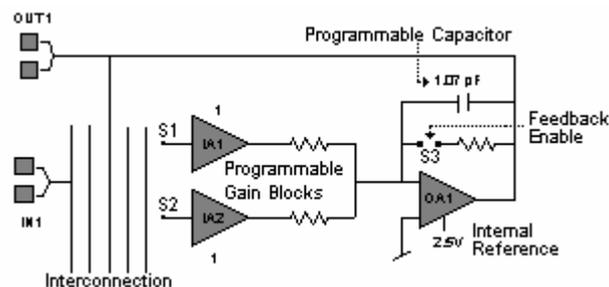
Figura 3.15: Representação em blocos dos componentes da família ispPAC da Lattice (LATTICE, 2000)

O CAB dos componentes da família ispPAC é baseado em transcondutores, conforme mostrado na Figura 3.16. O próprio fabricante representa os transcondutores por amplificadores de entrada (IA – *Input Amplifiers*) e resistores, com a intenção de facilitar a interpretação do circuito. Esta representação simplificada do CAB dos componentes da família ispPAC é mostrada na Figura 3.17.



**Figura 3.16:** Esquemático do CAB dos componentes da família ispPAC (LATTICE, 2000)

Considerando a representação da Figura 3.17, o CAB dos componentes da família ispPAC possui dois amplificadores de entrada com ganho programável entre  $-10$  e  $10$  um amplificador operacional de saída com dois laços de realimentação local. Um destes laços contém um resistor de valor fixo, porém habilitável, e outro contém um capacitor programável cujo valor pode variar entre  $1,06\text{pF}$  e  $61,59\text{pF}$ . Cada CAB, individualmente, pode implementar amplificadores somadores, filtros e integradores com e sem perdas. Os componentes dos CABs permitem a programação de filtros cuja frequência de corte pode ser programada entre  $10\text{ KHz}$  e  $595\text{ KHz}$ , aproximadamente. A largura de banda dos dispositivos da família ispPAC é  $650\text{ KHz}$  (LATTICE, 2000).



**Figura 3.17:** Representação do CAB dos componentes da família ispPAC (LATTICE, 2000)

Cada um dos FPAAs mostrados na Figura 3.15 apresenta diferenças quanto ao número de CABs e elementos adicionais. O ispPAC30 é o mais flexível dos três, pois o CAB representado na Figura 3.17 pode ser desmembrado, separando os IAs (*Input Amplifiers*) dos

OAs (*Output Amplifiers*). Desta maneira todos os 4 Ias podem ser interligados às entradas de um mesmo OA. Além disto, o ispPAC30 possui conversores D/A multiplicativos que também podem ser conectados aos OAs. Já o ispPAC20 possui dois CABs dois comparadores e um bloco de lógica digital simples que pode implementar uma porta XOR (*Exclusive OR*) ou um *flip-flop Set-Reset*. O ispPAC20 também possui um conversor D/A de 8 BITS que pode ser usado como um gerador interno ou externo de referência.

O componente considerado neste trabalho como um dos veículos de teste é o ispPAC10, pois dentre os membros da família ispPAC é que apresenta o maior número de CABs (quatro no total). A Figura 3.18 mostra a área de trabalho do software de programação dos componentes ispPAC com a representação global do componente ispPAC10. A ferramenta *PAC-Designer* (LATTICE, 2000), além de realizar a programação dos componentes, ainda possui um simulador que mostra a resposta em frequência dos circuitos programados nos FPAAs. O fato da ferramenta de simulação não apresentar a opção de simulação no domínio tempo ressalta a aplicação para a qual os componentes da família ispPAC são especializados: implementação de filtros de tempo contínuo. A Figura 3.19 mostra o resultado de uma simulação para um filtro cuja frequência de corte é programada com seu maior valor possível utilizando o FPAAs ispPAC10.

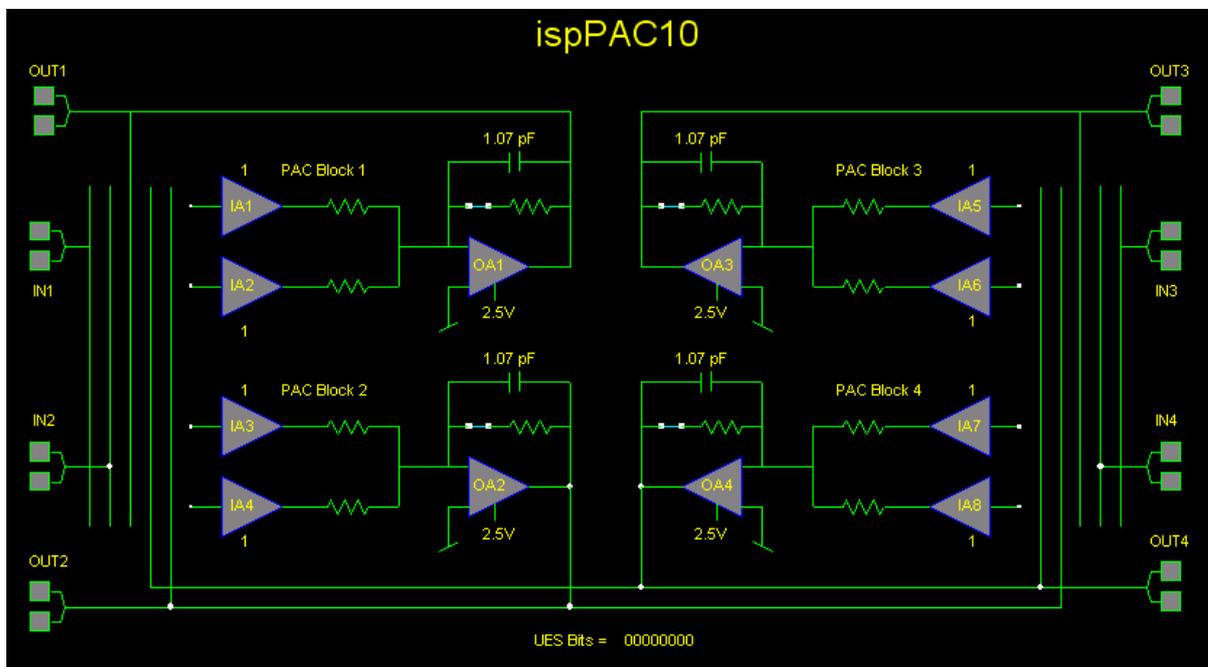


Figura 3.18: Representação global do FPAAs ispPAC10 (LATTICE, 2000)

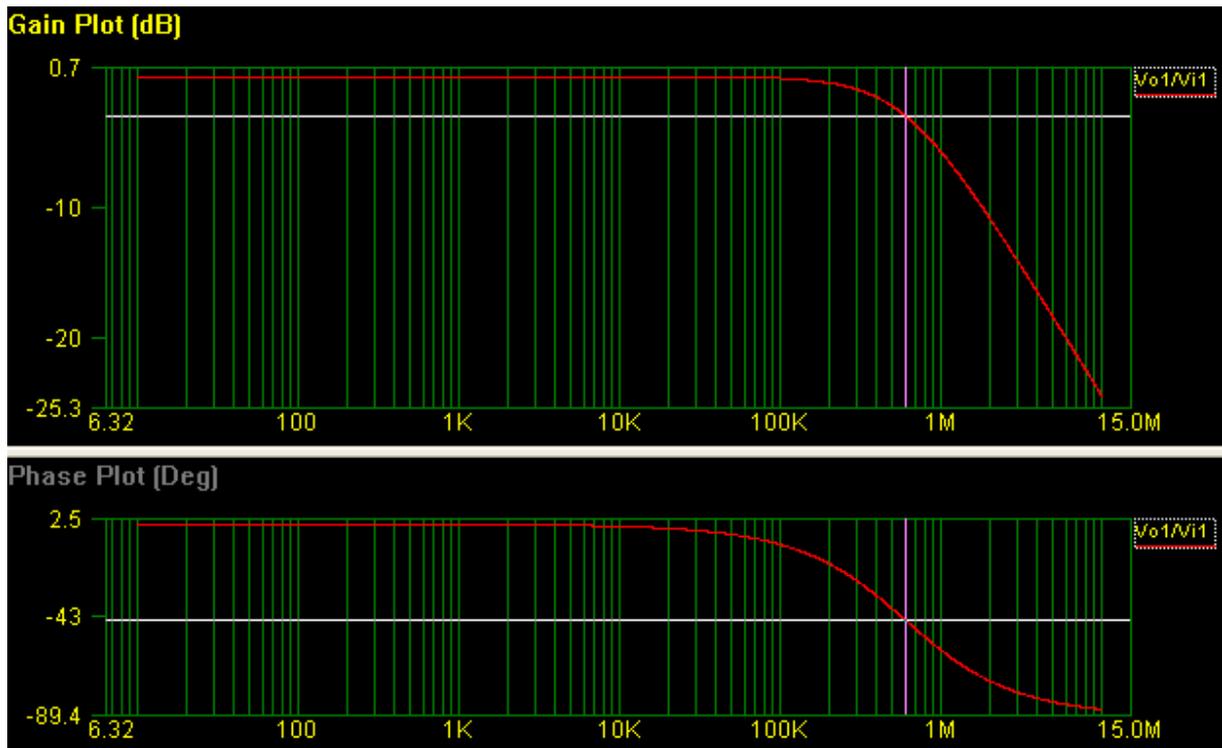


Figura 3.19: Resultado de uma simulação para um filtro programado com frequência de corte igual a 595KHz

Os componentes da família ispPAC possuem alimentação não simétrica de 5V. Por esta razão existe uma referência interna em 2.5V, porém, esta referência também pode ser programada para ser suprida externamente. Outra característica da família é a arquitetura totalmente diferencial dos blocos analógicos, conforme pode ser visualizado na Figura 3.16.

Demais características elétricas do FPAA ispPAC10 podem ser observadas no Anexo A.

### 3.3.2 O FPAA AN10E40 da *Anadigm Company*

Em 1996 a *Pilkinton Microelectronics*, situada na Inglaterra, desenvolvia um FPAA baseado na tecnologia de capacitores chaveados, denominado DPAD2 (BRATT; MACBETH, 1998). Uma versão posterior deste FPAA foi denominada DPAD3. Em 1997 a Motorola comprou a empresa inglesa e fundou o *Motorola Programmable Technologies Center*. O FPAA passou a ser comercializado pela Motorola com o nome de MPAA020 e pouco tempo depois, segundo (EETIMES, 2000), Macbeth deixou a Motorola após re-comprar a tecnologia do MPAA020 e, juntamente com Ludwig Klingenbeck, fundou a *Anadigm*.

São muitas as semelhanças entre o MPAA020 e o Atual AN10E40, ambos possuem uma matriz de 4x5 CABs, 13 células de I/O, têm o mesmo esquema de interconexões e finalmente, o software de programação *Anadigm Designer* é um melhoramento do *Easy Analog* da

Motorola, usado para a programação do MPAA020. Ainda, segundo (EETIMES, 2000), o hardware do atual AN10E40 é o mesmo do MPAA020. A representação por blocos do AN10E40 pode ser vista na Figura 3.20.

Este componente utiliza a tecnologia de capacitores chaveados implicando na amostragem do sinal analógico. A frequência de amostragem depende da frequência de relógio utilizada pelo FPAA. Segundo o manual do fabricante (ANADIGM, 2003) a máxima frequência de relógio comportada pelo dispositivo é 1 MHz o que de acordo com o teorema de Nyquist da amostragem (HAYKIN; VAN VEEN, 2001) limita a largura de banda deste dispositivo em 500 KHz.

Na Figura 3.20 é possível visualizar uma rede de interconexões circundando os 20 CABs. Esta rede é composta por um barramento vertical e um horizontal, totalizando 5 linhas e 6 colunas de interconexões, sendo que cada linha e coluna é composta por 2 fios. Este esquema de barramentos é definido como “interconexões globais” (BRATT; MACBETH, 1998), pois desta maneira é possível conectar um CAB a qualquer outro da matriz e a qualquer célula de I/O. As linhas do barramento vertical podem ser conectadas as linhas horizontais através de *cross over switches* (Figura 3.21).

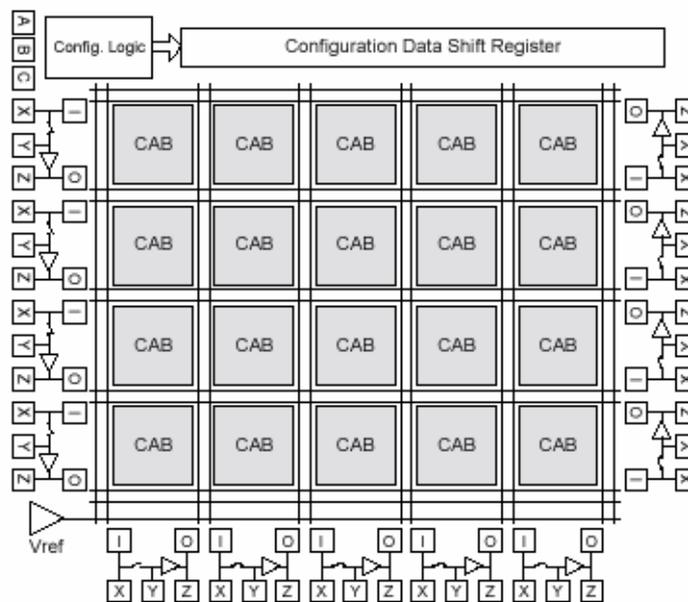
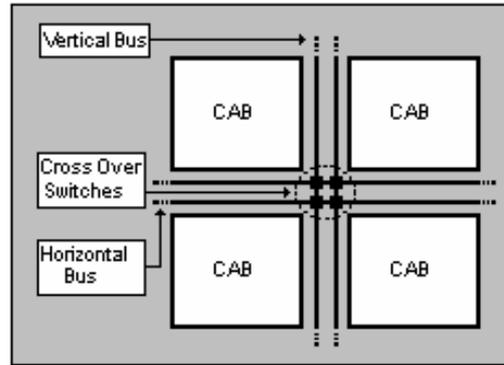
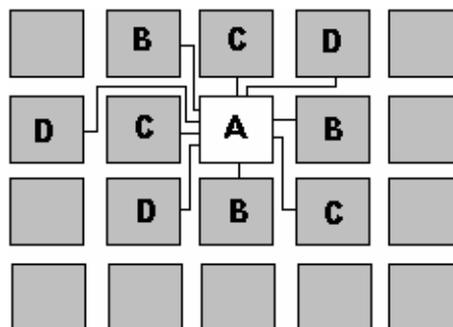


Figura 3.20: Representação em blocos do AN10E40 (ANADIGM, 2003)



**Figura 3.21** Barramentos globais e *cross over switches*

A matriz de interconexões globais não é o único recurso de roteamento do FPAA estudado. É possível conectar um CAB a outros CABs vizinhos utilizando interconexões locais. Estas conexões permitem que a saída de um CAB seja conectada à entrada dos 8 CABs que o circundam e do CAB situado duas posições à sua direita, além de sua própria entrada (BRATT; MACBETH, 1998). Tal esquema é muito útil em se tratando de dispositivos analógicos, pois os projetos analógicos usualmente são concebidos cascadeando-se blocos funcionais. Segundo (BRATT; MACBETH, 1998) cada CAB possui 5 conjuntos de entradas, denominadas entradas “A”, “B”, “C”, “D”, “E”. A entrada “A” é exclusiva da realimentação, ou seja, esta entrada só pode ser conectada à saída do seu próprio CAB. A entrada “E” também é especial, pois só pode ser conectada aos barramentos globais. As entradas “B”, “C” e “D”, são conjuntos de três entradas, que servem à conexão do CAB com seus 9 CABs vizinhos. A figura 3.22 mostra o esquema de conexões locais onde o CAB central (em branco), tem sua saída conectada à sua própria entrada A, enquanto os CABs que o circundam tem suas saídas conectadas nas entradas B, C e D do CAB central, conforme a letra que os representa. Com este esquema de interconexões locais é possível conectar as células mais próximas, deixando os barramentos globais livres, para que estes possam ser utilizados para conectar células mais distantes na matriz.



**Figura 3.22:** Conectividade local do AN10E40 (BRATT; MACBETH, 1998)

O CAB do AN10E40 é composto por um amplificador operacional, 5 bancos de capacitores e recursos de interconexão, conforme pode ser visualizado na Figura 3.23. A programabilidade do CAB é concebida através de chaves, cujo estado (aberta ou fechada) é definido pela memória de configuração, que é armazenada no *shift register* de programação. Cada componente programável ilustrado na Figura 3.23 apresenta um conjunto de chaves, estas chaves podem ser estáticas ou dinâmicas. As chaves estáticas são utilizadas para programar um dado valor de componente ou o estado de uma conexão e as chaves dinâmicas são utilizadas na implementação dos capacitores chaveados. A Figura 3.24 ilustra um banco de capacitores que pode ser programado como um capacitor ou como um resistor (capacitor chaveado) (ANADIGM, 2003).

No software de programação do componente AN10E40 existe uma biblioteca com 60 módulos que implementam as mais diversas funções analógicas, desde simples comparadores até os mais diversos tipos de filtros. Cada módulo possui um *bitstream* de programação próprio, que configura as chaves programáveis de maneira que o CAB execute a função requerida. Cada CAB possui aproximadamente 200 chaves programáveis (ANADIGM, 2003). O *bitstream* que programa um CAB possui 208 bits. Além destas chaves, o componente ainda conta com as chaves que programam as interconexões globais, as células de I/O e as lógicas de controle de relógio, de referência e de programação. No total são necessários 6864 BITS para programar todo o dispositivo (ANADIGM, 2003).

A alimentação do AN10E40 é não simétrica de 5V. Sua estrutura é totalmente diferencial. Mais detalhes sobre característica elétricas do componente podem ser encontrados no Anexo B.

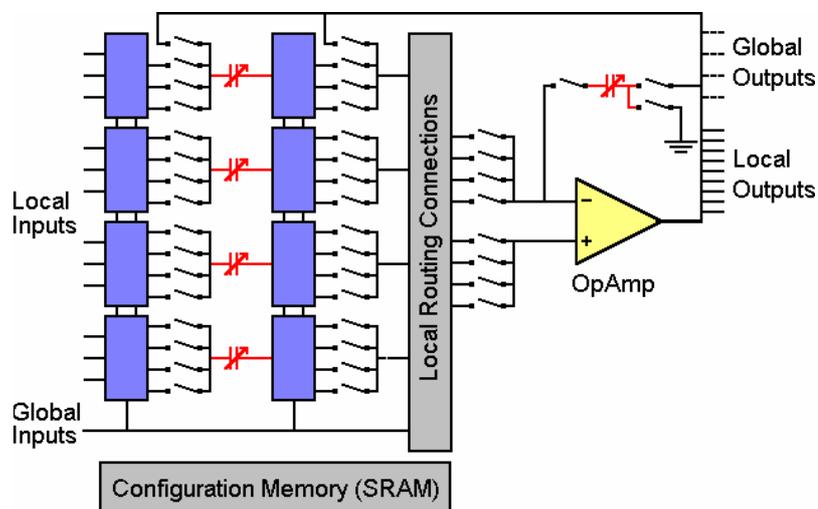


Figura 3.23: Representação do CAB do AN10E40 (ANADIGM, 2003)

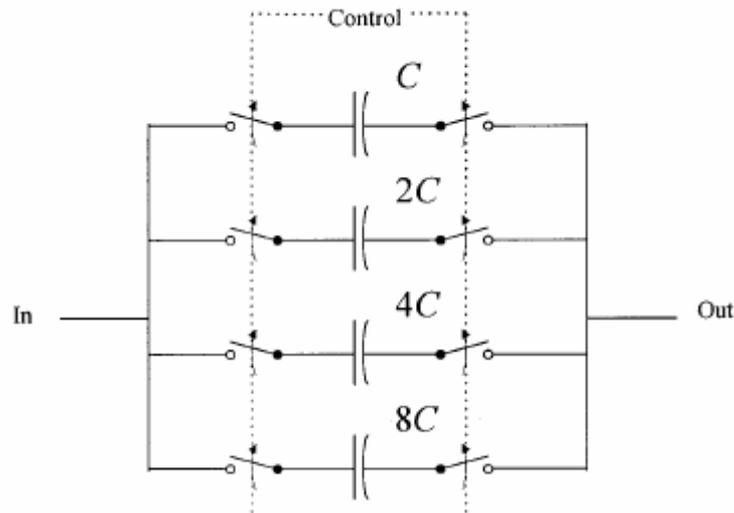


Figura 3.24: Banco programável de capacitores

### 3.3.3 O FPAA AN221E04 da *Anadigm Company*

O AN221E04 faz parte de uma família de componentes da segunda geração de FPAAs fabricados pela *Anadigm* (ANADIGM, 2003b). Em comparação com o componente da primeira geração, o AN10E40 (ANADIGM, 2003), o AN221E04 possui um menor número de CABs (4, contra 20 do componente da primeira geração). Contudo, os CABs do AN221E04 são mais versáteis e funcionais, sendo compostos por 3 elementos ativos (dois amplificadores operacionais e um comparador) enquanto o CAB do AN10E40 possui apenas um *OPAMP*. Adicionalmente, as células de entrada e saída (*I/O*) possuem recursos adicionais, como filtros *anti-aliasing* e *smoothing* e entradas multiplexadas permitindo o processamento de mais de um canal analógico pela mesma entrada.

São recursos extras também desta nova geração de componentes uma LUT (*Look Up Table*) e conversores A/D (analógico para digital), recursos estes que podem ser utilizados na linearização de sensores e processos de auto-calibração. A Figura 3.25 mostra o digrama em blocos do FPAA AN221E04 onde é possível visualizar os 4 CABs, as 4 células de entrada e saída, as duas células exclusivamente de saída, o esquema de interconexões, a LUT e outros blocos como geradores de referência e relógio.

Neste esquema é possível verificar que em cada CAB há uma área dedicada à memória de configuração do componente (SRAM) (ANADIGM, 2003b), diferentemente do AN10E40 no qual há apenas um *shift register* dedicado à programação de todo o componente. Tal

esquema demonstra uma das principais características desta nova geração de FPAA: a reconfiguração dinâmica. Nesta modalidade de programação, partes do circuito podem ser reconfiguradas durante o funcionamento do FPAA, enquanto outras partes continuam sua operação normal (ANADIGM, 2003b).

Assim como na geração anterior a tecnologia deste componente é CMOS e utiliza capacitores chaveados na implementação de resistores. A técnica de capacitores chaveados torna o circuito menos vulnerável a variações de processo e correntes parasitas (BRATT, 1998) e ainda desempenha importante papel na programabilidade do dispositivo. No entanto esta técnica impõe limitações à faixa de operação em frequência do componente, pois implica na amostragem do sinal analógico, demandando que a frequência do sinal a ser processado seja pelo menos a metade da frequência de chaveamento do dispositivo (teorema da amostragem de Nyquist) (HAYKIN; VAN VEEN, 2001).

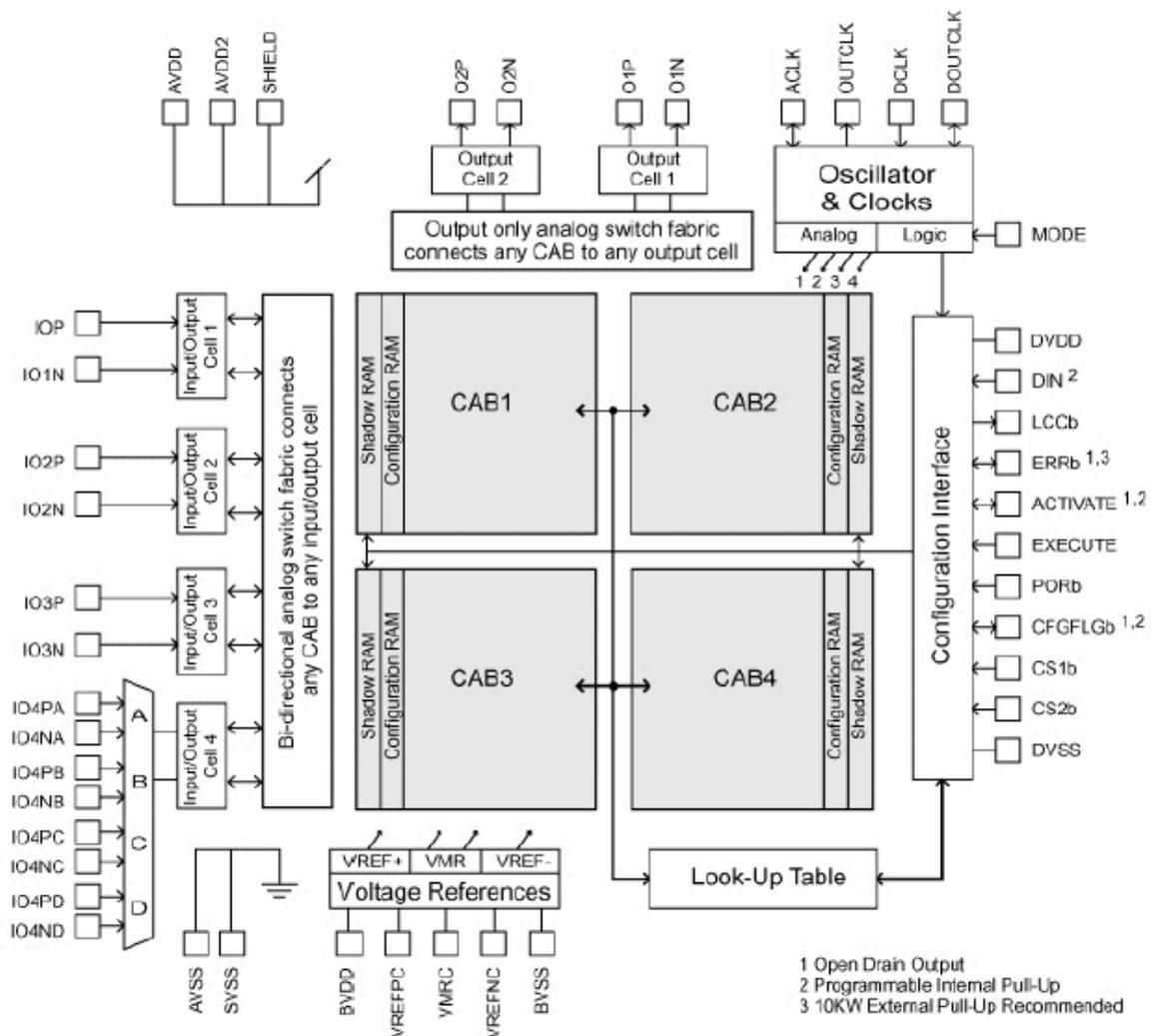
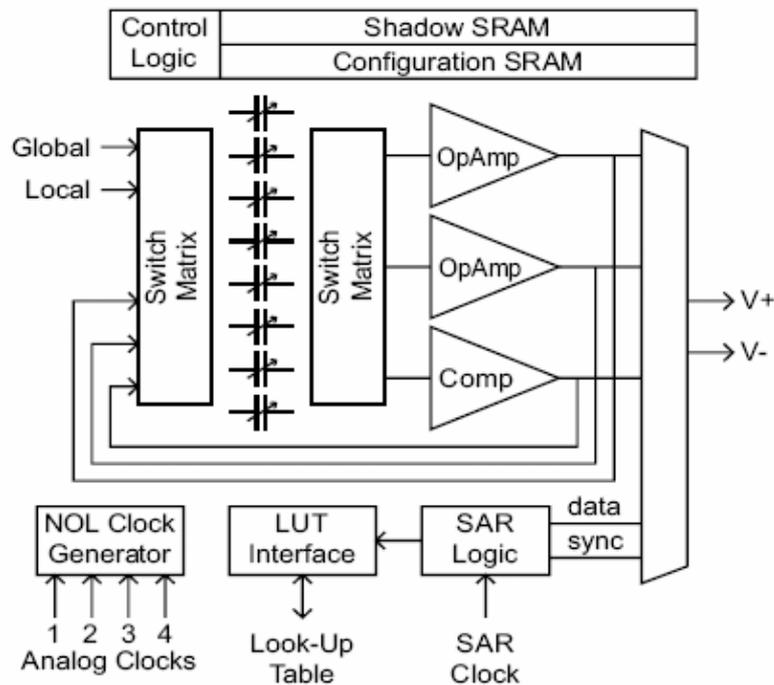


Figura 3.25: Diagrama em blocos do FPAA AN221E04. Extraído de (ANADIGM, 2003b)

Cada CAB do AN221E04 é composto por dois amplificadores operacionais, um comparador, blocos de interconexão, 8 bancos de capacitores programáveis, um conversor A/D por aproximações sucessivas (SAR – Successive Approximation Register) SAR-A/D, um bloco responsável pela geração dos sinais de relógio não sobrepostos (necessários ao funcionamento de circuitos a capacitor chaveado (SEDRA; SMITH, 1991)), além de sua memória de programação. O esquema de um CAB do FPAA AN221E04 pode ser visto na Figura 3.26.



**Figura 3.26:** Esquemático do CAB do FPAA AN221E04. Extraído de (ANADIGM, 2003b)

Segundo o manual do fabricante cada capacitor programável é composto por um banco “muito grande” de capacitores de igual tamanho, sendo que cada banco de capacitores pode assumir um valor relativo entre 0 e 255 unidades de capacitância (ANADIGM, 2003b). Através destas informações pode-se considerar um esquema para cada banco de capacitores como o mostrado na Figura 3.27, onde  $C_1 = C_2 = \dots = C_{255}$ .

A Figura 3.28(a) mostra a célula de entrada básica do AN221E04, a Figura 3.28(b) mostra a célula de entrada multiplexada e a Figura 3.29 mostra uma célula de saída.

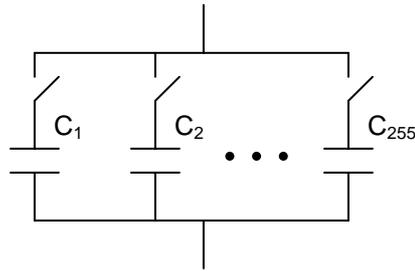


Figura 3.27: Esquemático considerado para os bancos de capacitores do AN221E04

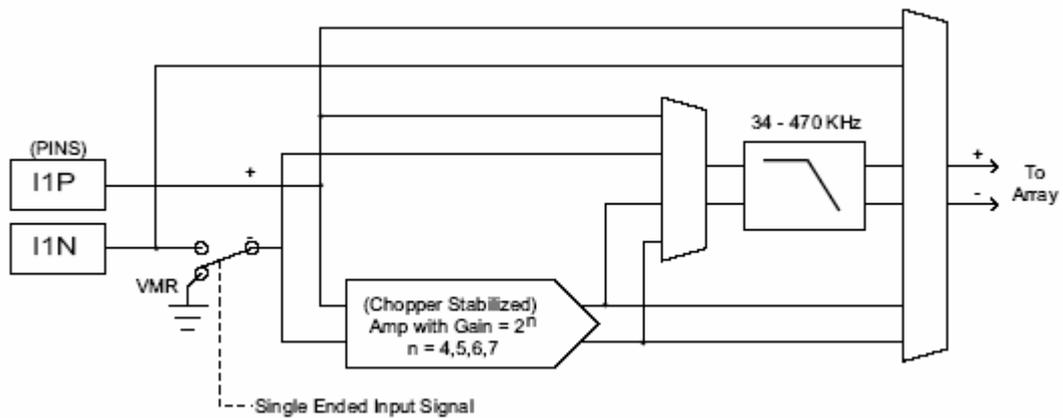


Figura 3.28(a): Célula de entrada do FPAA AN221E04. Extraído de (ANADIGM, 2003b)

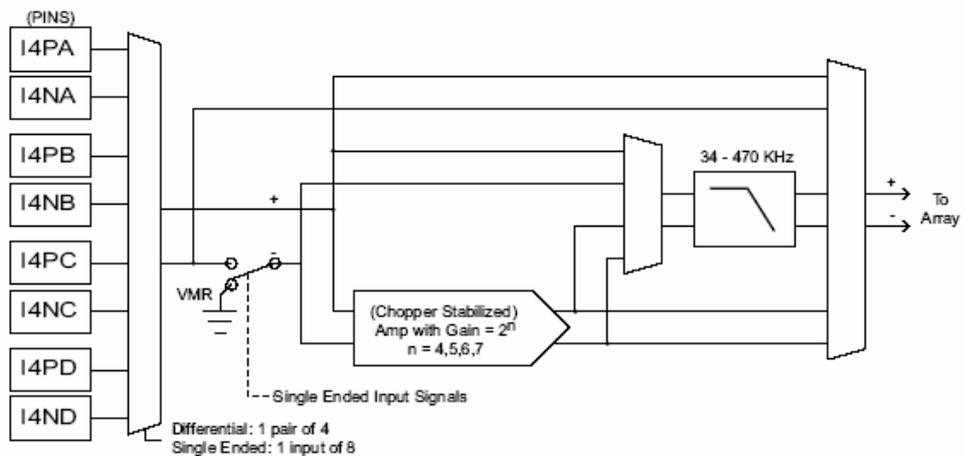


Figura 3.28(b): Célula de entrada multiplexada do FPAA AN221E04. Extraído de (ANADIGM, 2003b)

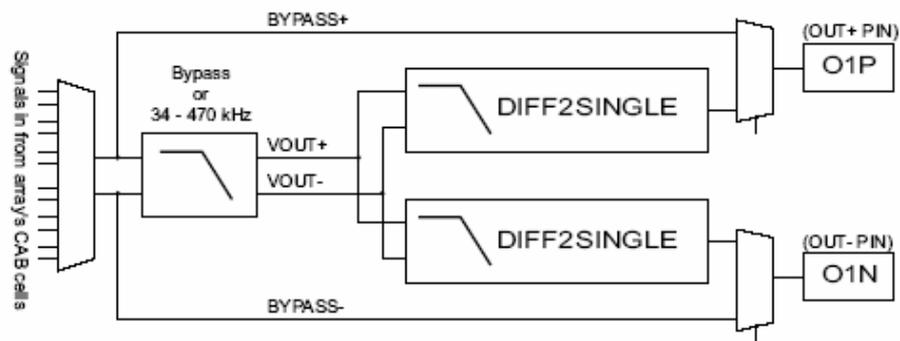


Figura 3.29: Célula de saída do FPAA AN221E04. Extraído de (ANADIGM, 2003b)

A programação deste FPAA pode ser feita através de um software específico fornecido pelo fabricante, denominado *Anadigm Designer II*. Esta ferramenta é uma interface gráfica de fácil utilização que permite escolher blocos programáveis disponíveis em uma biblioteca pré-concebida. A programabilidade do componente é restrita a parâmetros funcionais como frequência de corte e ganho, tomando como exemplo a programação de filtros. Por esta razão o usuário não tem acesso direto aos componentes internos de um CAB, e não pode alterar diretamente parâmetros como valores de resistores e capacitores, por exemplo. A Figura 3.30 ilustra este fato mostrando a tela da área de trabalho do software com um exemplo de programação dos parâmetros de um filtro.

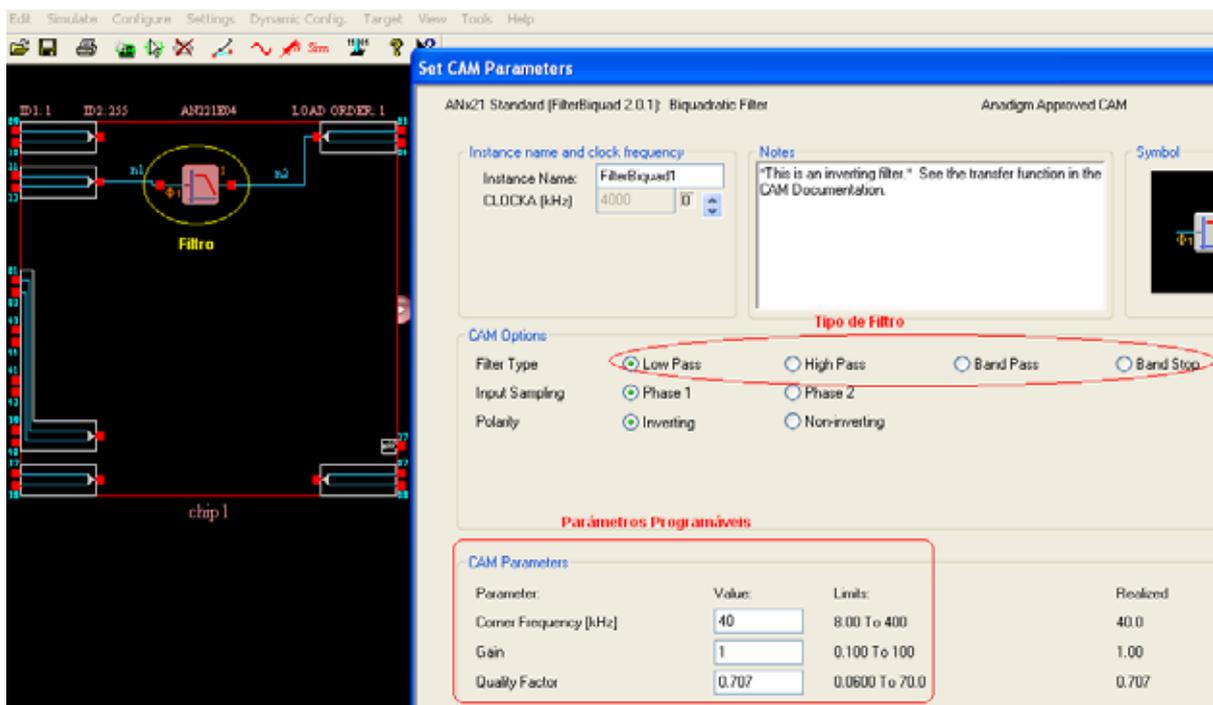


Figura 3.30: Exemplo de programabilidade com o software *Anadigm Designer II*

Assim como o AN10E40 a alimentação do A221E04 é não simétrica de 5V. Sua estrutura é totalmente diferencial. Mais detalhes sobre características elétricas do componente podem ser encontrados no Anexo C. Além do AN221E04 a segunda geração de FPAA's da *Anadigm*, denominada *Anadigm Vortex*, possui mais quatro componentes. A listagem destes componentes, bem como um comparativo entre eles são mostrados no Anexo D.

#### 4. TESTE DO FPAА ISPPAC10 DA LATTICE SEMICONDUCTORS

A programabilidade estrutural dos FPAAs da *Lattice* facilita o desenvolvimento e aplicação de metodologias também estruturais. O método de teste baseado em oscilação, já apresentado na seção 2.8, configura uma boa alternativa para este tipo de teste.

Em uma primeira abordagem foi considerada uma estratégia baseada no método OBT “puro”, ou seja, o CUT é transformado em osciladores e a análise dos parâmetros do sinal oscilatório é feita externamente. Na segunda abordagem um analisador de resposta (ORA: *Output Response Analyzer*) é construído internamente, utilizando-se os recursos do dispositivo com o intuito de facilitar a análise dos parâmetros do sinal gerado pelo oscilador e aumentar a cobertura de falhas.

As duas abordagens são aplicadas considerando os mesmos componentes e modelos de falhas, tendo como foco os CABs do dispositivo. Os resultados em termos de tempo de teste e cobertura de falhas são então comparados.

##### 4.1 OBT ASSOCIADO À ANÁLISE EXTERNA: BIST PARCIAL

A aplicação do OBT demanda que os blocos do circuito sejam, de alguma maneira, transformados em um ou mais osciladores. O componente considerado nesta etapa do trabalho, o ispPAC10, possui quatro CABs (LATTICE, 2000), sendo possível a implementação de até dois osciladores em uma única configuração do dispositivo.

Neste trabalho cada oscilador é construído utilizando-se dois CABs, em uma topologia conhecida como *quadrature ring oscillator* (ARABI; KAMINSKA, 1997), composta por dois integradores puros cascadeados e realimentados, conforme a Figura 4.1.

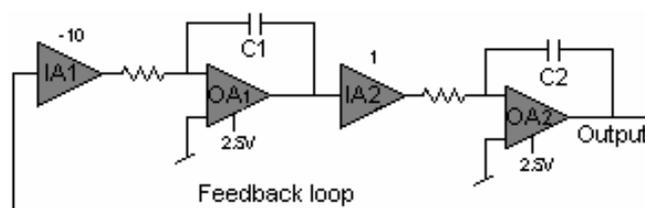


Figura 4.1: Esquemático do oscilador utilizado neste trabalho

Cada um dos integradores impõe ao sinal um desvio de fase de  $90^\circ$ , totalizando  $180^\circ$ . O ganho negativo do amplificador de entrada IA<sub>1</sub> representa mais um desvio de fase de  $180^\circ$ , totalizando  $360^\circ$ , o que satisfaz o critério da instabilidade de Barkhausen, conforme abordado anteriormente.

Do ponto de vista de sistema, é possível realizar uma análise quantitativa quanto ao diagrama de pólos e zeros deste oscilador. Segundo o manual do fabricante (LATTICE, 2000) a função de transferência de cada CAB é dada por:

$$V_{out} = \frac{k_1 \cdot g_m \cdot Vin_1 + k_2 \cdot g_m \cdot Vin_2}{g_{m3} + \frac{C}{2} \cdot S} \quad (11)$$

Onde  $g_m$  é o valor da transcondutância dos amplificadores de entrada (IAs),  $k$  é o ganho programado para o bloco,  $g_{m3}$  é o valor da transcondutância de realimentação,  $C$  é o valor do capacitor programável e  $Vin_1$  e  $Vin_2$  são os valores de tensão presente nos dois terminais de entrada do CAB.

Uma vez que o oscilador é construído utilizando-se apenas uma das entradas de cada CAB, um dos dois termos  $Vin_1$  ou  $Vin_2$  torna-se zero. Analogamente, como a transcondutância de realimentação não é utilizada na implementação do oscilador (veja Figura 4.1), o termo  $g_{m3}$  também é igual a zero. Assim, a função de transferência dos dois integradores cascadeados (considerando  $C1=C2=C$ ) pode ser expressa como:

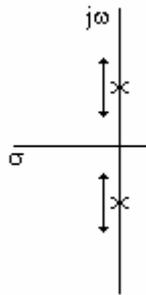
$$\frac{V_{out}}{V_{in}} = \frac{k_{1A} \cdot k_{1B} \cdot g_m^2}{\frac{C^2}{4} \cdot S^2} \quad (12)$$

Onde os índices “A” e “B” referenciam o ganho dos IAs dos diferentes CABs (A e B) utilizados na implementação dos integradores. Considerando a realimentação negativa imposta pelo ganho negativo do IA<sub>1</sub> é possível obter a função de transferência do oscilador aplicando a Equação 3:

$$\frac{V_{out}}{V_{in_{osc}}} = \frac{k_{1A} \cdot k_{1B} \cdot g_m^2}{\frac{C^2}{4} \cdot S^2 + k_{1A} \cdot k_{1B} \cdot g_m^2} \quad (13)$$

Resolvendo a equação característica (polinômio do denominador) obtêm-se duas raízes complexas conjugadas, o que significa que os pólos da função de transferência estão sobre o eixo  $j\omega$ , conforme a representação da Figura 4.2. Teoricamente isto significa que obtêm-se

uma oscilação não amortecida cuja frequência (localização dos pólos) depende dos termos  $C$ ,  $KI_A$ ,  $KI_B$  e  $gm$ .



**Figura 4.2: Pólos do oscilador localizados sobre o eixo  $j\omega$**

Na prática, devido a não idealidades do circuito, é difícil obter os pólos exatamente sobre o eixo  $j\omega$  sendo necessário um estímulo para que o circuito comece a oscilar. Por esta razão optou-se por programar um ganho elevado na entrada do bloco oscilador (-10, conforme pode ser visto na Figura 4.1). Este ganho elevado amplifica o ruído e a tensão de *offset* presentes na entrada do circuito, atuando como o estímulo necessário para iniciar a oscilação. Desta maneira, a oscilação é garantida para todos os valores programáveis de capacitância do FPAA, o que não acontece quando o módulo do ganho do laço é unitário, por exemplo (dados experimentais).

Após a programação dos osciladores no FPAA a assinatura do circuito sem falhas é computada, ou seja, para cada valor programável de capacitância, considerando  $C_1=C_2$  e mantendo o ganho dos blocos fixo (conforme a Figura 4.1), são medidos os valores da amplitude e da frequência de oscilação. Durante o teste os valores obtidos para estes parâmetros são comparados com os valores de assinatura para um circuito sem falhas, considerando uma banda de tolerância. Se a assinatura obtida durante o teste estiver fora da banda de tolerância o componente sob teste é considerado falho.

Assim, é possível programar quantos osciladores forem necessários para cobrir todos os CABs do FPAA e testá-los em paralelo, diminuindo o tempo de teste. Para cobrir os 4 CABs do ispPAC10 dois osciladores são construídos, permitindo o teste de todos os CABs em uma única configuração estrutural. A Figura 4.3 mostra a área de trabalho do software *PAC-Designer* com a representação dos osciladores programados.

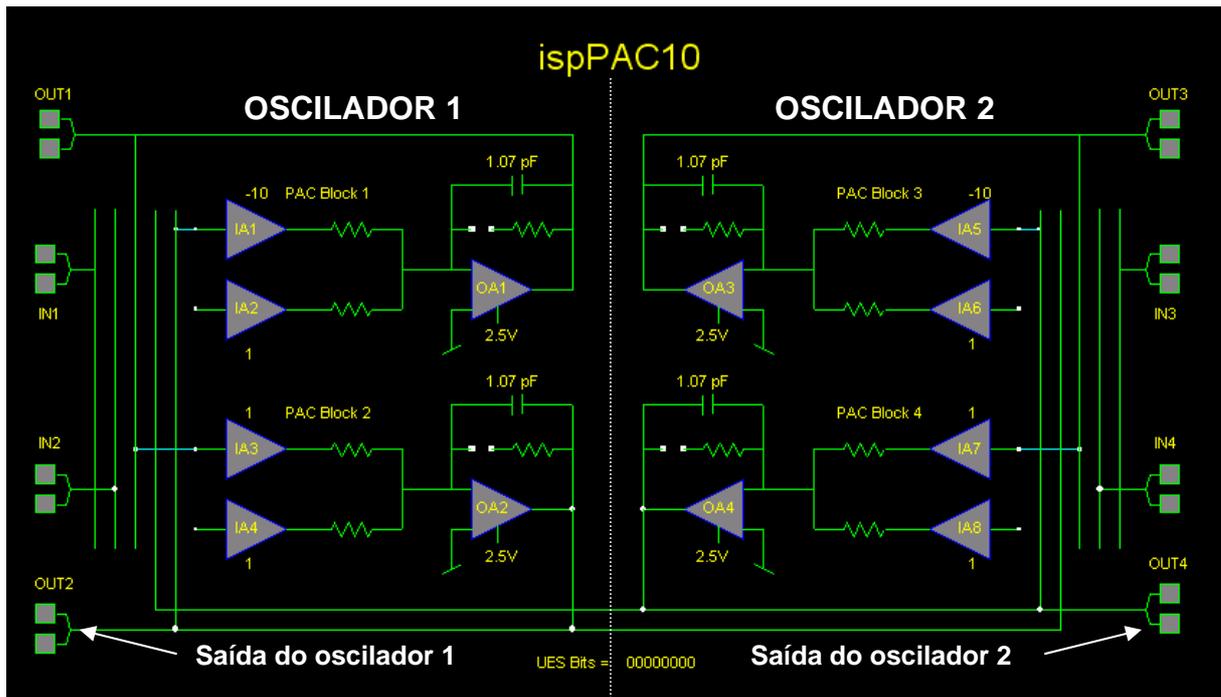


Figura 4.3: Representação dos osciladores na área de trabalho do software *PAC-Designer*

Para estimar a cobertura de falhas que o método pode alcançar, quando aplicado a este FPAA, uma série de experimentos de injeção de falhas é realizada. As falhas são injetadas utilizando-se o próprio software de programação do dispositivo, configurando os componentes de maneira a simular desvios paramétricos em relação aos valores dos componentes sem falhas. No caso das chaves programáveis as falhas são injetadas programando-as para operarem de maneira inversa, *i. e.*, se o comportamento correto depende da chave permanecer fechada esta é programada para permanecer aberta e *vice-versa*. Este método de injeção de falhas limita-se às possibilidades de programação disponíveis para o FPAA considerado.

#### 4.1.1 Teste dos Capacitores Programáveis

O primeiro passo para os teste dos capacitores é obter a assinatura do oscilador livre de falhas para todos os valores programáveis possíveis dos capacitores. Para facilitar o teste os dois capacitores do oscilador são programados com o mesmo valor ( $C_1=C_2$ ). Os valores programáveis de capacitância variam entre 1,07pF e 61,59pF em passos discretos, totalizando 127 valores diferentes para cada capacitor. Todas as configurações possíveis (considerando o valor dos capacitores) são programadas no FPAA, utilizando a placa de desenvolvimento do

ispPAC10. Para cada configuração a frequência de oscilação e a amplitude do sinal são medidas e computadas como o comportamento sem falhas do circuito.

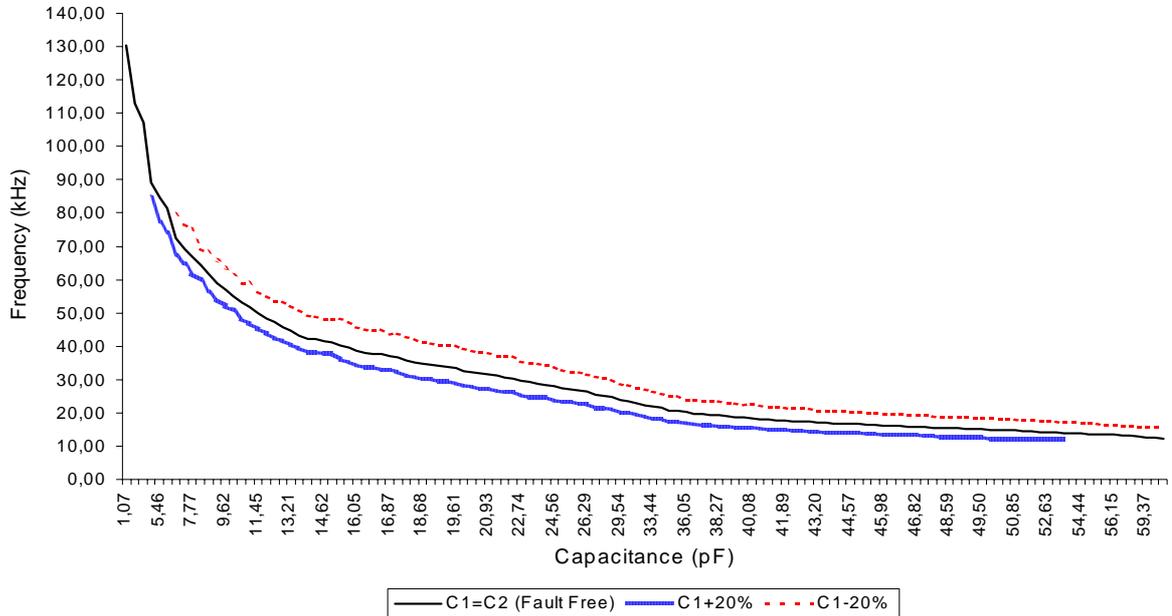
Respeitando as possibilidades de programação do dispositivo, desvios de  $\pm 20\%$  nos valores nominais dos capacitores são considerados neste modelo de falhas. Segundo (ALLEN; HOLBERG, 1987) a variabilidade típica nos valores de capacitores CMOS é aproximadamente 10%.

Uma vez que os valores de capacitância programáveis são discretos nem sempre é possível injetar desvios de  $\pm 20\%$  exatamente. Logo, para cada uma das falhas consideradas, admite-se uma variação de  $\pm 5\%$ , ou seja, as falhas injetadas variam entre  $\pm 15\%$  a  $\pm 25\%$ . Fora desta faixa de variação considera-se que não é possível injetar a falha, e, portanto, esta não é considerada nas estimativas de cobertura de falhas e tempo de teste. Exemplificando: para os valores de capacitância entre 1,07pF a 3,11pF (3 valores), não é possível injetar desvios paramétricos sem sair da faixa de variação considerada; para dois valores, 5,06pF e 5,46pF, apenas variações positivas foram injetadas; já para a faixa entre 53,93pF e 61,59pF (11 valores) apenas os desvios negativos podem ser injetados. Na faixa remanescente (entre 5,92pF e 53,53pF, totalizando 111 valores) tanto desvios positivos como negativos podem ser injetados.

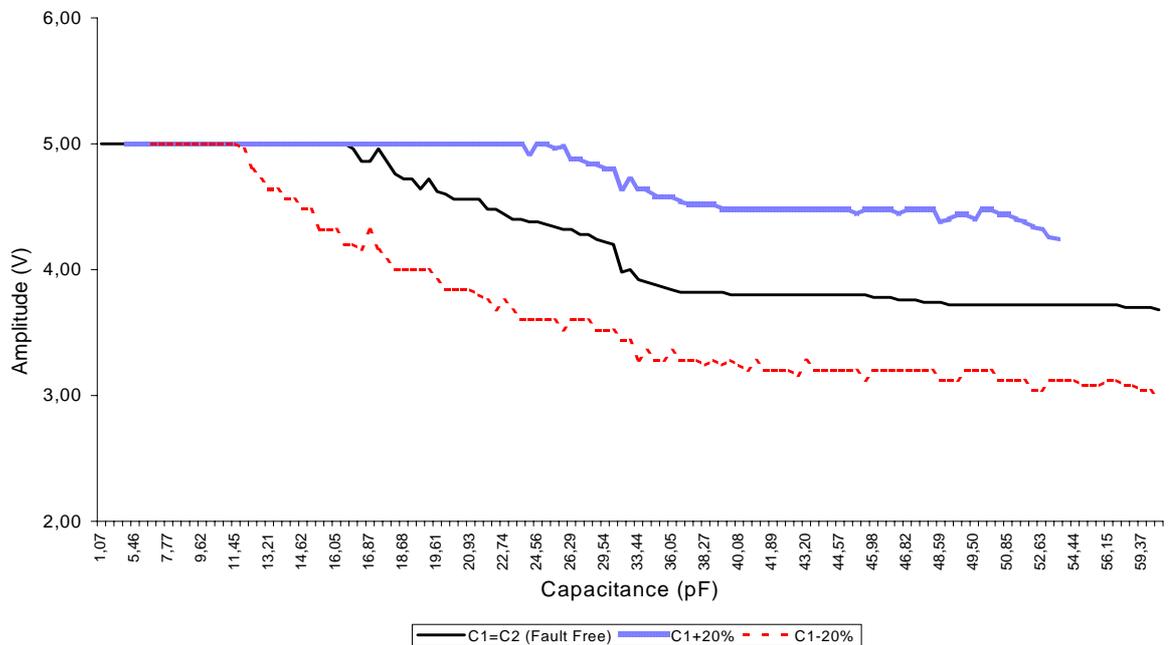
Desta maneira, 940 falhas foram injetadas nos capacitores programáveis (2 falhas  $\times$  111 valores programáveis, mais 1 falha  $\times$  13 valores programáveis, para cada um dos 4 CABs), modificando o valor nominal dos capacitores através do software de programação. As Figuras 4.4 e 4.5 mostram, respectivamente, os valores medidos de frequência e amplitude em função dos valores dos capacitores programáveis, considerando o circuito sem falhas e com os desvios do modelo de falhas adotado. Os resultados mostrados dizem respeito ao capacitor  $C_1$  (capacitor do primeiro integrador de um dos osciladores). Os resultados obtidos para o capacitor  $C_2$  (segundo integrador) e para os capacitores  $C_3$  e  $C_4$  (do segundo oscilador) são análogos.

Comparando os resultados obtidos é possível concluir que, neste caso, a frequência de oscilação é o parâmetro mais apropriado para detecção de falhas na faixa de capacitância que vai desde os menores valores até os valores intermediários. De modo complementar, a amplitude do sinal oscilatório permite a detecção de falhas de maneira facilitada quando considerada a faixa de capacitância que compreende os valores intermediários até os valores mais elevados. Assim, a observação dos dois parâmetros assegura uma cobertura de falhas de

100%, considerando o modelo adotado para os capacitores programáveis. Para o teste dos capacitores, segundo a estratégia descrita acima, é necessário configurar o FPAA 124 vezes.



**Figura 4.4:** Frequência de oscilação para o circuito sem falhas e considerando desvios de  $\pm 20\%$  nos capacitores programáveis



**Figura 4.5** Amplitude de oscilação para o circuito sem falhas e considerando desvios de  $\pm 20\%$  nos capacitores programáveis

#### 4.1.2 Teste dos Blocos de Ganho (IAs)

O teste dos blocos de ganho do ispPAC10 utilizando o método baseado em oscilação apresenta um problema particular. Para os altos valores de ganho observa-se uma baixa sensibilidade da frequência e da amplitude de oscilação em relação a variações no ganho. Com o intuito de minimizar este problema uma série de experimentos práticos foi realizada com a intenção de se obter uma configuração na qual a sensibilidade seja maximizada. Foi observado que a maior sensibilidade da frequência de oscilação a variações no ganho acontece quando os capacitores são programados com o seu valor mínimo (1,07pF). A amplitude não é considerada nesta etapa, pois não acrescenta informação no que diz respeito à observabilidade das falhas consideradas.

O ganho do primeiro CAB de cada oscilador é fixo em -1 e o ganho do outro CAB é variado de +1 a +10. Para garantir a oscilação os ganhos dos dois blocos do oscilador necessitam ter sinais opostos. Logo, para testar o ganho do segundo bloco no intervalo de valores que vai de -1 a -10 o ganho do primeiro bloco deve ter seu valor modificado para +1. Observa-se que para valores baixos de ganho a sensibilidade da frequência de oscilação é elevada e diminui à medida que o valor do ganho aumenta. Para valores de ganho maior do que 5 em módulo a sensibilidade é pequena, o que dificulta a observação de falhas.

Uma vez que os valores programáveis de ganho variam apenas em passos inteiros o modelo de falhas para os blocos de ganho fica restrito também a variações inteiras, tal que  $K_{falha} = K \pm 1$ . Onde  $K$  representa o ganho e  $K_{falha}$  representa o ganho na presença de uma falha. As mesmas limitações na injeção de falhas para os valores programáveis extremos acontecem quando considerado o ganho. Assim sendo, não é possível injetar falhas  $K + 1$  quando o valor do ganho é -1 ou 10. Da mesma maneira que não é possível injetar falhas  $K - 1$  para valores de ganho igual a 1 e -10. Por isto, o número total de falhas injetas para cada IA é 36. Como o componente possui 8 IAs, são injetadas 288 falhas (36x8) segundo o modelo adotado.

As curvas que representam o comportamento da frequência de oscilação em relação a variações no ganho, para o circuito com e sem falhas, são mostradas na Figura 4.6. Nesta Figura é possível observar a baixa sensibilidade da frequência para os valores de ganho maiores que 5. Considerando uma banda de tolerância para a frequência de oscilação,  $\Delta f_{min} = 500Hz$ , dentro da qual uma falha permaneceria não detectada, apenas as falhas para  $|K| \leq 5$  são observáveis.

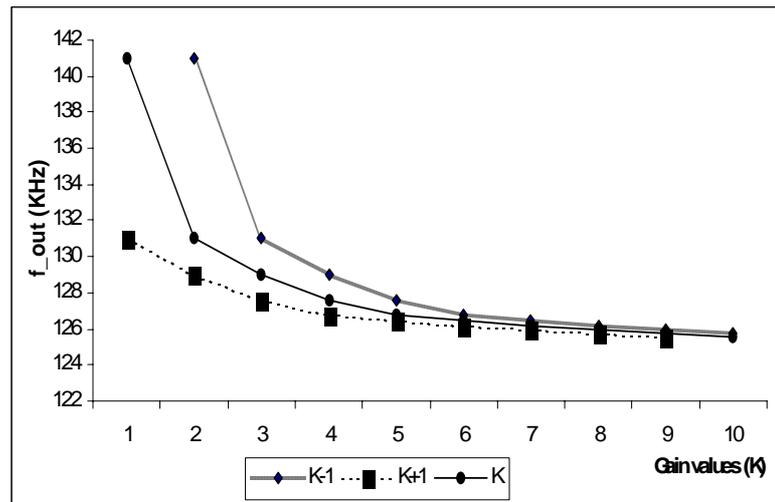


Figura 4.6: Comportamento da frequência em relação ao ganho para os circuitos com e sem falha

Cada CAB dos osciladores possui 2 IAs, por isso, é necessário repetir o teste do oscilador para contemplar os 4 IAs. Os outros dois CABs que compõem o segundo oscilador podem ser testados em paralelo. Uma vez que se sabe que esta estratégia não cobre falhas para os valores de ganho maiores que 5 estes valores não necessitam ser configurados durante o teste. Desta maneira, o número configurações de teste necessárias para cobrir todos os blocos de ganho é 40 no total. Devido à baixa sensibilidade da frequência de oscilação observada para metade dos valores programáveis de ganho, apenas 144 das 288 falhas são detectadas. Logo, a cobertura de falhas obtida para os IAs utilizando o método OBT associada a análise externa de resposta é 50%.

#### 4.1.3 Teste das Chaves Programáveis dos CABs

Testar as chaves programáveis de um CAB utilizando o método OBT é uma tarefa simples, pois uma chave aberta no caminho do sinal pode fazer com que o circuito pare de oscilar. Já se a chave que habilita a realimentação do *OPAMP* permanecer fechada, o integrador deixa de ser puro, inibindo a oscilação.

Foram consideradas 3 chaves compondo cada CAB: as chaves  $S_1$  e  $S_2$  nas entradas dos amplificadores  $IA_1$  e  $IA_2$  e a chave  $S_3$  que habilita a realimentação local do *OPAMP* do CAB (conforme pode ser visto na Figura 4.7). De fato, as entradas  $IA_1$  e  $IA_2$  podem ser conectadas a diferentes linhas de interconexões internas e pinos externos, no entanto considera-se que as demais chaves fazem parte da rede de interconexões do dispositivo, por isso não são consideradas neste trabalho.

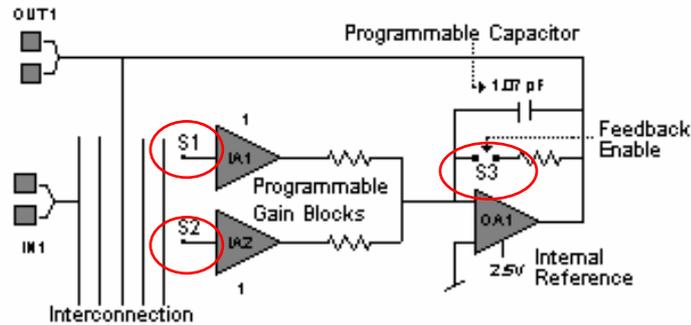


Figura 4.7: Chaves programáveis do CAB do ispPAC10

Para testar as chaves  $S_1$  e  $S_2$  cada CAB do oscilador é configurado de maneira a utilizar uma das duas entradas ( $IA_1$  ou  $IA_2$ ). Para testar a outra chave a configuração é modificada para que o IA que anteriormente não havia sido utilizado faça parte do laço do oscilador. Então, se uma falha do tipo *stuck-open* (sempre aberta) afetar uma das chaves de entrada o circuito não oscilará. Para detectar falhas do tipo *stuck-on* (sempre fechada) o procedimento é análogo, porém, as chaves são mantidas abertas e o comportamento esperado é a ausência de sinal oscilatório. Se houver a observação de um sinal oscilatório detecta-se uma falha do tipo *stuck-on* em uma destas chaves. Para testar a chave  $S_3$  o procedimento é análogo, porém, ao contrário das chaves  $S_1$  e  $S_2$ , a ausência de oscilação é determinada por uma falha do tipo *stuck-on*.

A Tabela 1 mostra as configurações necessárias para testar as chaves  $S_1$ ,  $S_2$  e  $S_3$ , bem como o comportamento esperado para cada configuração. Durante o teste de uma chave as demais permanecem com a sua configuração original.

Tabela 1: Configurações para o teste das chaves

Falha	Config. das chaves	Comportamento esperado	Comportamento com falhas
$St\_on\ S1$	$S1, S2, S3\ off$	Ausência de oscilação	Oscilação
$St\_on\ S2$	$S1, S2, S3\ off$	Ausência de oscilação	Oscilação
$St\_on\ S3$	$S1\ on, S2, S3\ off$	Oscilação	Ausência de oscilação
$St\_open\ S1$	$S1\ on, S2, S3\ off$	Oscilação	Ausência de oscilação
$St\_open\ S2$	$S1, S3\ off, S2\ on$	Oscilação	Ausência de oscilação
$St\_open\ S3$	$S1, S3\ on, S2\ off$	Ausência de oscilação	Oscilação

De acordo com os dados da Tabela 1, é possível concluir que são necessárias 4 diferentes configurações para testar todas as chaves de um CAB. Como o teste é idêntico para o outro CAB, mais 8 configurações são necessárias. Os outros dois CABs que compõem o segundo oscilador são testados concorrentemente, não aumentando o número de

configurações de teste. As estimativas de tempo de teste e cobertura de falhas são mostradas na seção 4.3.

#### **4.2 OBT ASSOCIADO À ANÁLISE INTERNA DE RESPOSTA: BIST COMPLETO**

A estratégia de BIST completo desenvolvida para o teste do FPAA ispPAC10 mescla o método de teste baseado em oscilação, descrito na seção 2.8, e a análise de resposta baseada em duplo integrador (ABILBO), apresentado na seção 2.7.

Com os 4 CABs disponíveis no componente é possível construir duas estruturas ABILBO. Cada uma destas estruturas pode operar como um gerador de sinal de teste (oscilador em quadratura) ou um analisador de resposta (ORA) baseado em dupla integração (LUBASZEWSKI; MIR; PULZ, 1996). A implementação de um oscilador em quadratura no ispPAC10 foi descrita na seção 4.1, onde a aplicação do método OBT associado à análise externa foi apresentada. Na estratégia descrita nesta seção um ORA baseado em dupla integração é utilizado para compactar a resposta de teste (sinal gerado pelo oscilador), facilitando a análise de resposta e até aumentando a cobertura de falhas.

Dois dos quatro CABs do ispPAC10 são utilizados na implementação de um oscilador, de maneira idêntica à apresentada na seção 4.1. Os dois CABs restantes são então utilizados para construir um duplo integrador, que desempenha o papel de analisador de resposta, conforme a Figura 4.8. O comparador do ORA é implementado externamente e a tensão de referência utilizada é 4V. A saída do oscilador é conectada a entrada do ORA.

O analisador de resposta proposto em (LUBASZEWSKI; MIR; PULZ, 1996) é composto por dois integradores puros. Neste trabalho uma modificação foi feita na topologia do ORA: o analisador considerado é composto por um integrador com perdas e um integrador puro, conforme pode ser visto na Figura 4.8. O motivo desta modificação foi a baixa repetibilidade das assinaturas obtidas para os baixos valores de capacitância do oscilador, conforme publicado em (BALEN *et al.*, 2004). Observou-se durante os experimentos práticos que o sinal resultante da dupla integração (com os dois integradores puros) consistia de um sinal parabólico, análogo aos representados na Figura 4.9, enquanto o esperado seria uma rampa somada a um sinal oscilatório. Além disto, a saída do integrador saturava rapidamente o que dificultava a observabilidade das assinaturas.

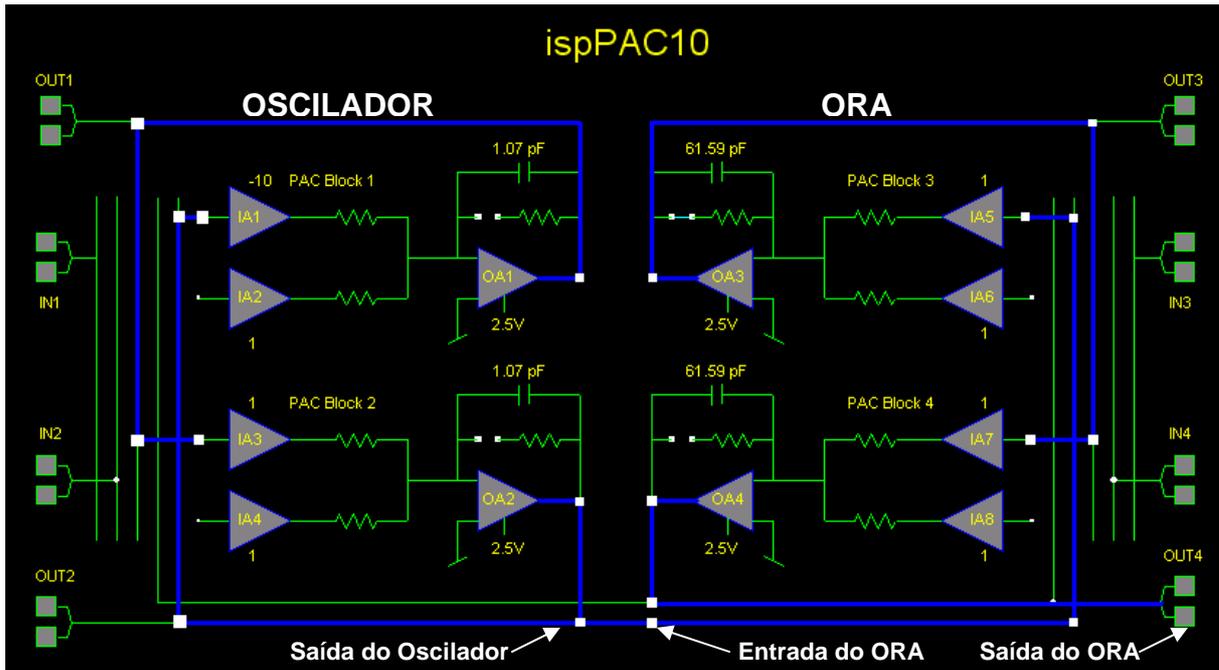


Figura 4.8: Representação do oscilador e do ORA baseado em duplo integrador

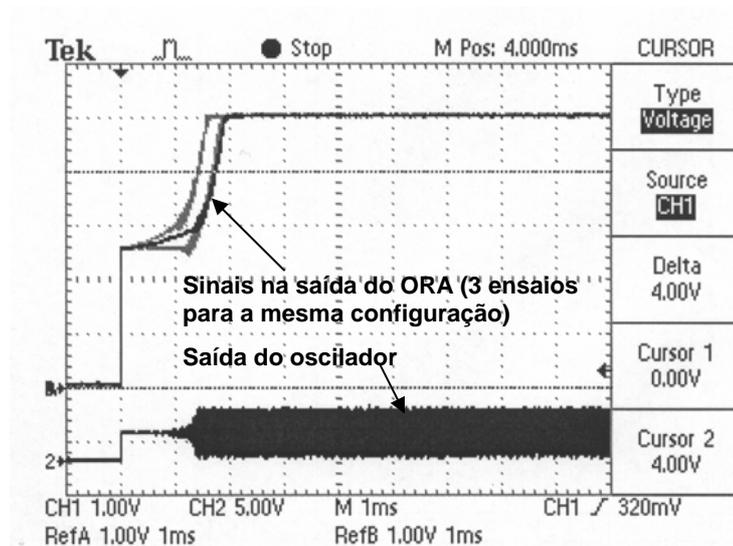


Figura 4.9: Sinal parabólico observado na saída do ORA (integradores puros) em três ensaios para uma mesma configuração

Uma hipótese levantada para o surgimento deste sinal parabólico foi de que a saída do oscilador apresentaria uma tensão de *offset* não nula em relação à entrada do ORA. Esta tensão DC seria integrada uma vez pelo primeiro integrador, gerando uma rampa, que integrada novamente geraria a parábola observada. A amplitude do sinal oscilatório somado à parábola, neste caso, é pequena, pois o sinal na saída do ORA começa a aumentar rapidamente, antes mesmo do estabelecimento da amplitude do sinal oscilatório (Figura 4.9).

O estabelecimento por completo da oscilação pode demorar tempos diferentes em diferentes ensaios, pois o início da oscilação depende de variáveis que podem diferir de um ensaio para outro, tais como o ruído na entrada do oscilador e a temperatura. Uma vez que a saída do ORA satura rapidamente quando utilizado os dois integradores puros, conforme a Figura 4.9, pode-se dizer que a maior parte do sinal integrado é justamente o início da oscilação. Tal fato seria uma hipótese para explicar as diferentes assinaturas obtidas para uma mesma configuração.

Com a adição de uma realimentação DC no primeiro integrador, transformando-o em um integrador com perdas, observou-se que a saída do ORA não mais apresenta um comportamento parabólico. A repetibilidade das assinaturas também melhorou consideravelmente. A Figura 4.10 mostra a saída do ORA para 3 ensaios de teste (sobre uma mesma configuração) quando o integrador com perdas é utilizado. É possível observar na Figura 4.10 a tendência esperada para o sinal de saída do ORA (rampa somada a sinal oscilatório) bem como a alta repetibilidade das assinaturas obtidas.

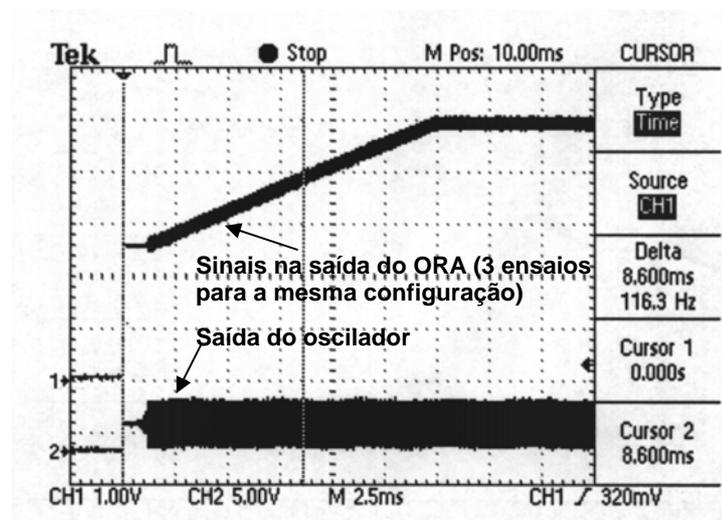


Figura 4.10: Sinais adquiridos durante ensaios de teste utilizando um integrador com perdas no ORA

#### 4.2.1 Análise de Resposta para o Teste dos Capacitores Programáveis

Para diminuir o número de configurações de teste necessárias para cobrir todos os capacitores programáveis do FPAA, considerou-se a possibilidade de testar não apenas os capacitores do oscilador, mas também os capacitores dos integradores (em uma mesma configuração do dispositivo). Porém, os experimentos mostraram que para valores menores que 30pF, falhas nos capacitores dos integradores apresentam uma baixa observabilidade. Logo para estes valores de capacitores apenas os CABs que implementam o oscilador são

testados ao mesmo tempo, enquanto os CABs que compõem o ORA são mantidos fixos e com a máxima constante de tempo programável dos integradores ( $C_3=C_4=61,59pF$ ) uma vez que o aumento da constante de tempo facilita a detecção de falhas (LUBASZEWSKI; MIR; PULZ, 1996). Posteriormente a configuração do dispositivo é modificada, de maneira que os CABs que no primeiro momento foram utilizados na implementação do ORA sejam agora utilizados na construção do oscilador. Desta maneira todos os CABs do FPAA são testados.

Conforme já mencionado, o tempo de *start up* da oscilação é um parâmetro que também interfere no teste. Quando utilizado o integrador com perdas no ORA a variação deste tempo para uma mesma configuração não é significativa a ponto de prejudicar o teste. Contudo, este parâmetro é sensível a variações nos componentes dos CABs, como valores dos capacitores e do ganho dos IAs. Por isso, o tempo de início da oscilação pode ser considerado como um quarto parâmetro de teste (além da amplitude, frequência e nível DC). Para tirar vantagem deste parâmetro adicional é preciso garantir que o tempo de *start up* não seja insignificante na assinatura de tempo obtida pelo ORA. Também se espera que este não seja dominante, pois desta maneira pequenas variações no *start up* para uma mesma configuração poderiam ser detectadas como variações na assinatura, aumentando a probabilidade de *fault aliasing* (quando análise de um padrão de teste induz a detecção de uma falha que não existe, ou quando uma falha deixa de ser detectada. Isto ocorre devido a variações que ocasionem outra falha ou desvio que “esconda” a falha em questão). O tempo de *start up* é inversamente proporcional ao ganho total do laço do oscilador, ou seja, quanto maior é o ganho menor é o tempo necessário para a oscilação se estabelecer completamente. Por isso, para diferentes faixas de valores dos capacitores, o ganho de laço do oscilador é configurado com diferentes valores, obtendo tempos de *start up* adequados. O tempo de *start up* pode ser visualizado nos sinais mostrados na Figura 4.10, onde é possível observar um tempo de aproximadamente 1.4ms antes do estabelecimento integral da oscilação.

Dados os pontos colocados cima, três conjuntos de teste são realizados para obter os melhores resultados no teste dos capacitores. Para a faixa de capacitâncias entre 1pF e 13pF o módulo do ganho de laço programado para o oscilador é 2 e as constantes de tempo do ORA são fixadas no máximo valor possível. Para os valores de capacitores entre 14pF e 30pF as constantes de tempo continuam fixas no máximo valor e o módulo do ganho de laço é 10. Para valores de capacitores maiores que 30pF o módulo do ganho de laço é mantido elevado ( $|G|=10$ ), porém, a constante de tempo dos integradores é variada de maneira que os valores dos capacitores do ORA sejam iguais aos valores dos capacitores do oscilador, pois para esta

faixa de capacitâncias, falhas nos capacitores do ORA também são detectadas. A Figura 4.11 ilustra as assinaturas sem falha e com um desvio de +20% no capacitor  $C_1$  (do oscilador) quando este é programado em 10pF.

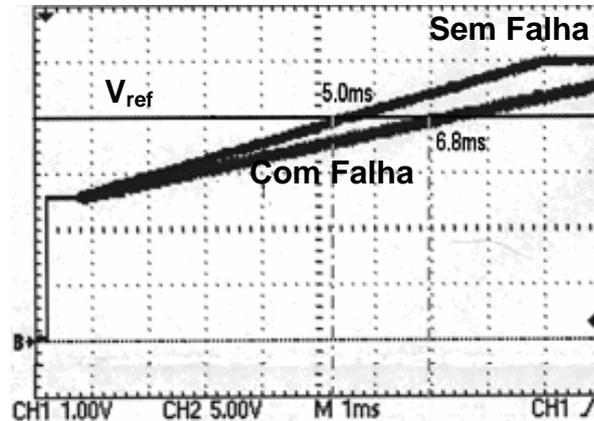


Figura 4.11: Assinaturas de tempo sem falhas e para um desvio de +20% no capacitor  $C_1$  do oscilador

A Figura 4.12 mostra os resultados obtidos para o teste dos capacitores, na faixa de valores inferior a 30pF, sob a forma de um histograma. Desvios de +20% e -20% foram injetados. Foi considerado que para uma falha ser detectada a assinatura obtida para esta falha teria que diferir no mínimo 0,3ms da assinatura nominal. Esta tolerância foi derivada experimentalmente por se tratar da maior variação observada para uma mesma configuração em diferentes ensaios. As colunas do histograma marcadas em preto representam as falhas não detectadas.

As mesmas restrições quanto à injeção de falhas descritas anteriormente ocorrem nestes experimentos. Logo, para esta faixa de capacitores, exclui-se do teste os três primeiros valores totalizando 59 capacitores a serem testados. Como é preciso repetir o procedimento de teste para testar os capacitores dos CABs que primeiramente foram utilizados no ORA, um total de 118 (2x59) configurações de teste se faz necessário para a faixa de capacitores menores que 30pF.

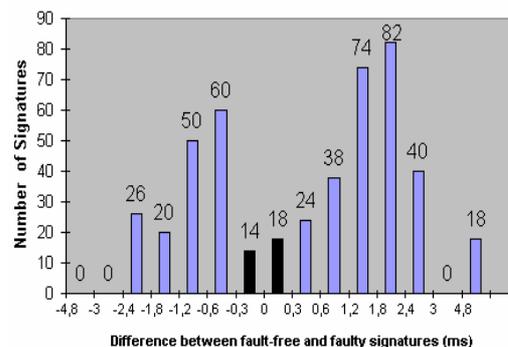


Figura 4.12: Histograma das diferenças entre as assinaturas de tempo para os circuitos com e sem falhas nos capacitores programáveis considerando os valores menores que 30pF

Os melhores resultados obtidos com esta metodologia de teste foram observados na terceira e última faixa de valores de capacitância (31pF a 61,59pF). Para esses valores o ORA é mais sensível a falhas nos capacitores dos seus próprios integradores, permitindo que os quatro capacitores (tanto os do oscilador quanto os do ORA) sejam testados concorrentemente. Uma vez que existem 65 valores de capacitância nesta faixa, é necessário configurar o FPAA mais 65 vezes para testar completamente todos os capacitores do dispositivo. Os resultados obtidos para esta faixa de capacitâncias podem ser visualizados nas curvas das Figuras 4.13 a 4.16, que mostram respectivamente as assinaturas de tempo, com e sem falhas, observadas durante o teste dos capacitores  $C_1$  e  $C_2$  (do oscilador), e  $C_3$  e  $C_4$  (do duplo integrador). Devido às restrições quanto a injeção de falhas no extremo superior dos valores de capacitores, para a faixa entre 55pF e 61,59pF (11 valores), a assinatura do circuito com falhas não é mostrada, pois não é possível injetar desvios de +20% através da técnica de injeção utilizada neste trabalho. Como pode ser visto nas Figuras, desvios ocasionados nas assinaturas de tempo são suficientemente expressivos para permitir a detecção das falhas consideradas no modelo proposto.

A Tabela 2 mostra o número de falhas injetadas, cobertas e não cobertas, bem como o número de configurações necessárias à aplicação dessa estratégia de BIST para cada faixa de capacitância.

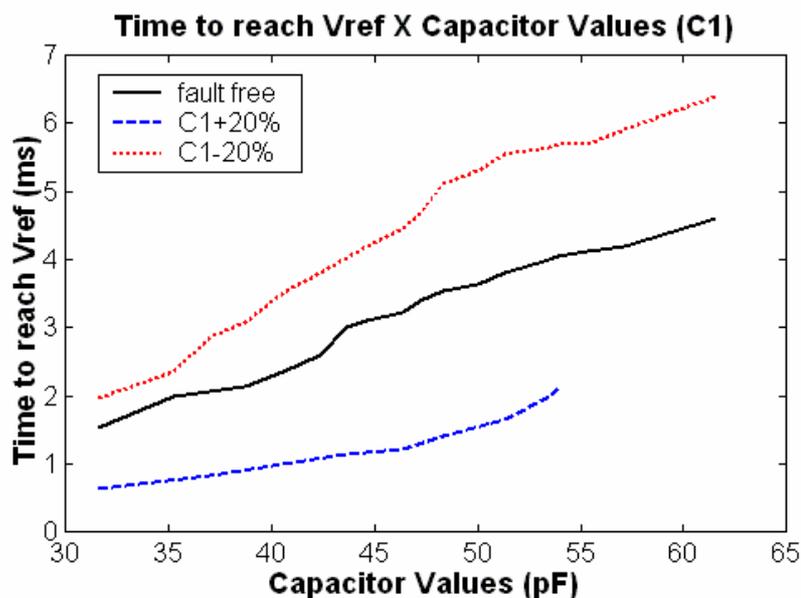


Figura 4.13: Assinaturas com e sem falhas no capacitor  $C_1$  do oscilador para valores de capacitâncias maiores que 30pF

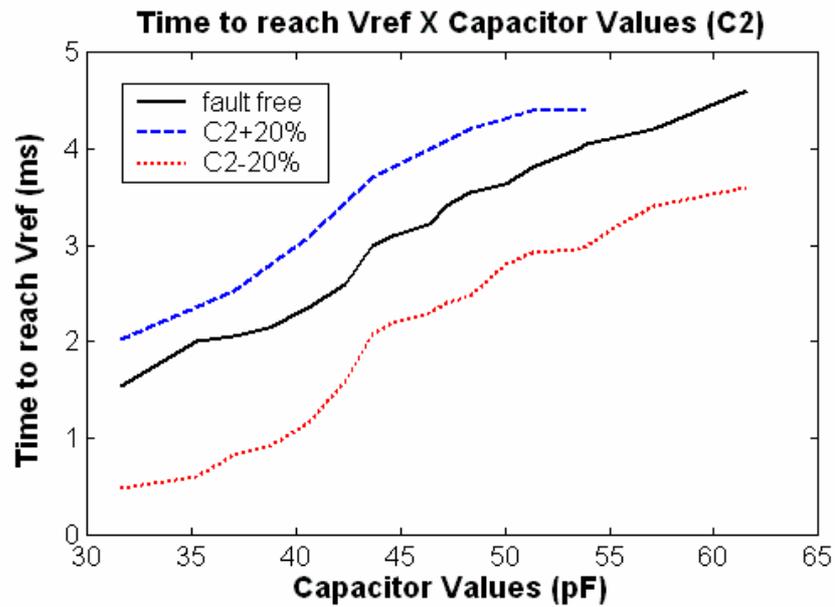


Figura 4.14: Assinaturas com e sem falhas no capacitor  $C_2$  do oscilador para valores de capacitâncias maiores que 30pF

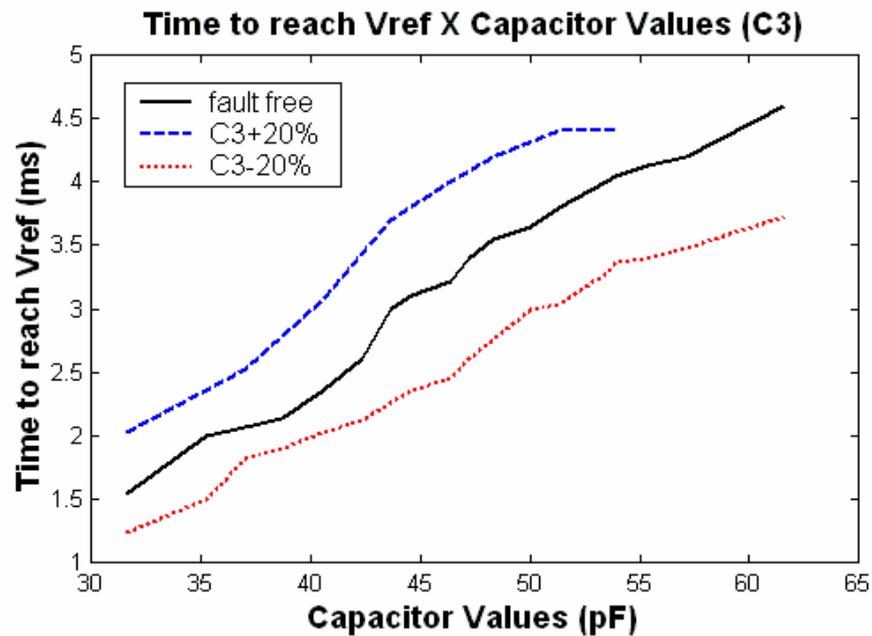


Figura 4.15: Assinaturas com e sem falhas no capacitor  $C_3$  do oscilador para valores de capacitâncias maiores que 30pF

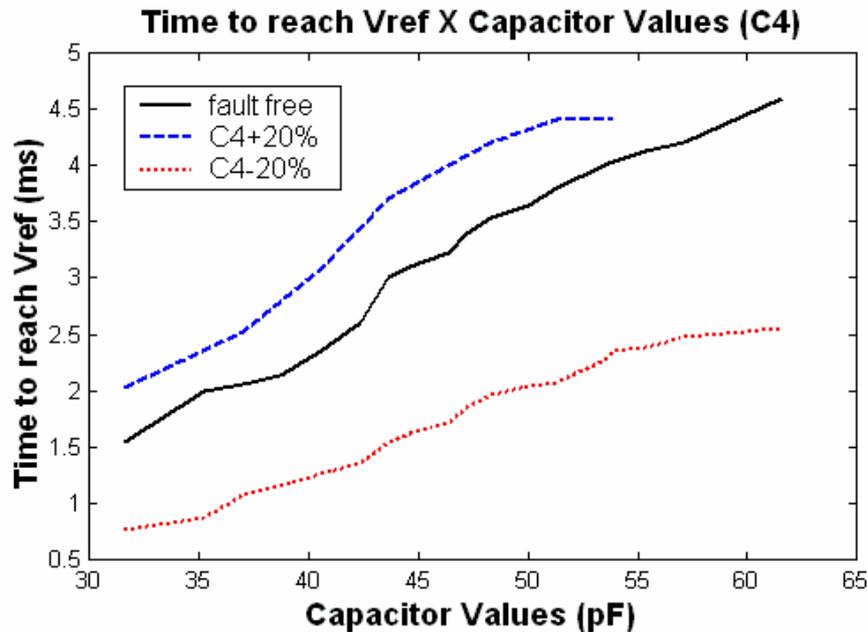


Figura 4.16: Assinaturas com e sem falhas no capacitor  $C_4$  do oscilador para valores de capacitâncias maiores que 30pF

Tabela 2: Cobertura de falhas e número de configurações para o teste dos capacitores utilizando o ORA

Faixa de capacitância	Valores programados	Falhas injetadas	Falhas cobertas	Não cobertas	Cobertura de falhas	Configurações
1,07pF a 30pF	62	464	432	32	93,1%	118
31pF a 61,59pF	65	476	476	0	100%	65
Total	127	940	908	32	96,6%	183

#### 4.2.2 Análise de Resposta para o Teste dos Blocos de Ganho (IAs)

Para o teste do ganho utilizando o analisador de assinatura (ORA), uma estratégia diferente daquela apresentada na seção 4.1.2 é adotada. Conforme citado anteriormente, a assinatura de tempo dada pelo ORA é dependente do tempo de *start up* da oscilação. Este parâmetro, por sua vez, apresenta uma boa sensibilidade a variações no ganho de laço do oscilador. Assim, é possível obter vantagem da sensibilidade do *start up* a variações no ganho para melhorar a cobertura de falhas obtida no teste dos IAs.

Assim como no teste dos capacitores, é preciso tomar cuidado com o valor programado do ganho de laço do oscilador, pois valores elevados podem acarretar em um curto tempo de *start up*, diminuindo a sensibilidade da assinatura de tempo a variações no ganho. Por outro

lado, baixos valores de ganho podem ocasionar tempos de *start up* elevados, o que aumenta o tempo de teste significativamente. Por isso, para testar cada IA do oscilador, o ganho do IA sob teste é variado ao longo de toda escala programável e o ganho do segundo IA é programado de maneira que o ganho total de laço do oscilador seja o mais próximo possível de 10. Uma vez que o ganho só pode ser programado em passos discretos unitários, para alguns valores de ganho do IA sob teste o ganho de laço nem sempre pode ser exatamente 10, variando entre 7 e 12.

O melhor compromisso entre sensibilidade e tempo de teste observado nos experimentos práticos é obtido quando os capacitores  $C_1$  e  $C_2$  são programados com os seus máximos valores. Desta maneira, ao mesmo tempo em que se garante a oscilação, a sensibilidade da assinatura do ORA ao tempo de *start up* é assegurada e o tempo de teste se mantém dentro de um limite aceitável. Os IAs do ORA são programados com ganho unitário positivo e os capacitores dos dois integradores são programados com seus máximos valores (61,59pF).

Considerando o  $IA_1$  como bloco sob teste e o segundo IA do oscilador como sendo o  $IA_3$ , conforme a Figura 4.8, 10 configurações são necessárias para testar o  $IA_1$  (uma para cada valor de ganho programável positivo). O valor do ganho do amplificador  $IA_3$  é variado de maneira a garantir que o ganho total de laço permaneça no intervalo descrito acima. Dado que este esquema também é capaz de detectar falhas no amplificador  $IA_3$ , não há a necessidade de configurar o componente mais 10 vezes, mas apenas 4. Isto ocorre porque durante o teste do  $IA_1$  o ganho do  $IA_3$  foi programado com 6 valores diferentes para respeitar a restrição considerada para o ganho total de laço. Para testar os valores negativos de ganho o mesmo esquema é utilizado, porém os sinais dos IAs são invertidos. Desta maneira, com 28 configurações de teste é possível testar 2 IAs dos CABs que compõem o oscilador. Uma vez que o FPAA dispõe de 4 pares de IAs no total, 112 configurações são necessárias para testar todos os blocos de ganho do dispositivo.

O mesmo modelo de falhas descrito na seção 4.1.2 ( $K_{falha} = K \pm 1$ ) foi utilizado. A Figura 4.17 mostra as curvas das assinaturas de tempo obtidas para variações no ganho  $IA_1$ . O aspecto não monotônico destas curvas deve-se ao fato do ganho de laço variar no intervalo entre 7 e 12, conforme citado anteriormente. Assim, para os diferentes valores de ganho (eixo horizontal), o ganho total de laço é diferente, o que faz as assinaturas de tempo variarem. A Figura 4.18 mostra um histograma com o número de ocorrências de desvios nas assinaturas do circuito com e sem falhas, considerando as falhas injetadas nos IAs. Diferentemente do teste baseado em oscilação com análise externa (seção 4.1), a cobertura de falhas obtida é

100%. A razão para esta melhoria na cobertura de falhas dos blocos de ganho é a sensibilidade do ORA ao tempo de *start up* do oscilador (que é dependente do ganho), enquanto o OBT com análise externa apenas leva em conta a frequência e a amplitude do sinal oscilatório.

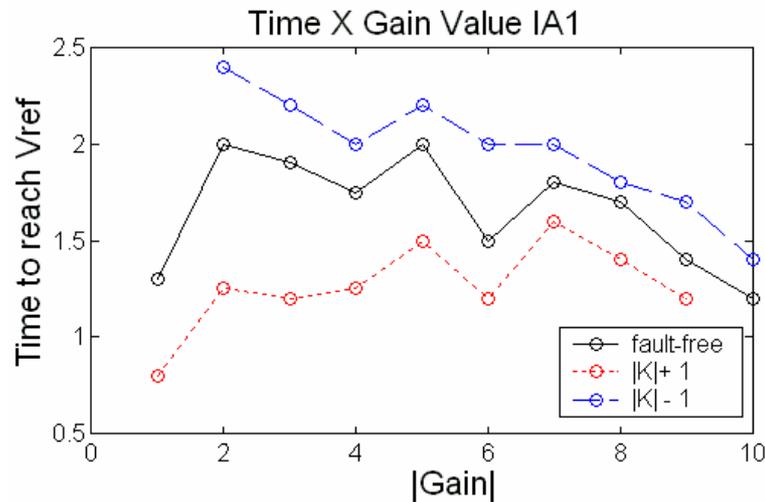


Figura 4.17: Assinaturas com e sem falhas para o teste do bloco de ganho IA<sub>1</sub>

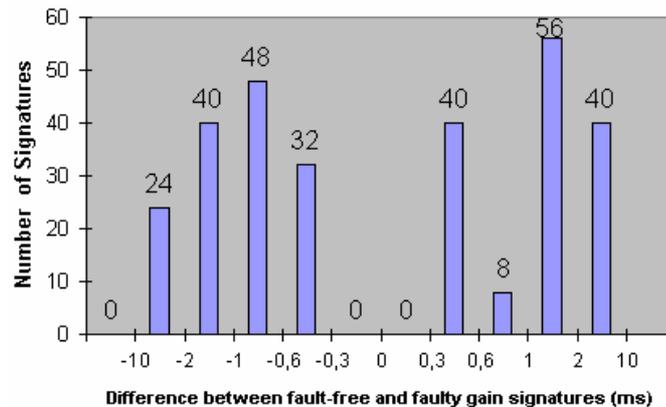


Figura 4.18: Histograma das diferenças entre as assinaturas de tempo para os circuitos com e sem falhas nos blocos de ganho segundo o modelo  $K_{falha} = K \pm 1$

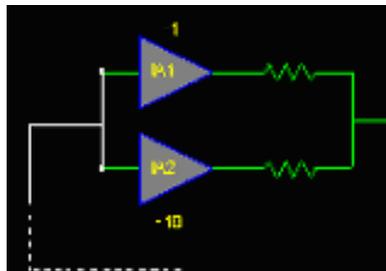
### 4.2.3 Análise de Resposta para o Teste das Chaves Programáveis dos CABs

De acordo com a Tabela 1 existem duas possibilidades para o comportamento do CUT no caso de uma das chaves presentes no caminho do sinal apresentar uma falha: ou circuito oscila, ou não oscila. Para detectar uma falha observando o comportamento do sinal na saída do oscilador basta saber qual é a configuração das chaves programáveis.

Para uma falha do tipo *stuck-on* na chave  $S_3$  não há oscilação e a entrada do ORA é então submetida à tensão de referência interna do FPAA. No caso da existência de um sinal oscilatório uma assinatura de tempo é obtida. Logo, as assinaturas com e sem falhas são facilmente discriminadas.

Se a ausência de oscilação é devida a um *stuck-open* em  $S_1$  ou  $S_2$  a assinatura de tempo obtida na saída do ORA é não nula e não apresenta repetibilidade. Trabalhou-se com a seguinte hipótese para esta falta de repetibilidade: se uma das chaves  $S_1$  ou  $S_2$  permanecerem abertas o IA correspondente terá sua entrada desconectada do circuito, e, dado que a impedância de entrada dos IAs é elevada (LATTICE, 2000), este IA torna-se então mais susceptível ao ruído. Segundo a hipótese levantada, seria então o ruído na entrada de um dos IAs, devido ao seu caráter aleatório, o responsável pela falta de repetibilidade nas assinaturas.

Levando em conta esta hipótese, uma nova estratégia de teste das chaves foi desenvolvida consistindo em utilizar como entrada do CAB os dois IAs em paralelo. Desta maneira, se uma falha simples ocorrer em um dos IAs desconectando sua entrada, o outro IA permanece conectado ao laço do oscilador, evitando que as duas entradas fiquem flutuando. A utilização dos dois IAs em paralelo como entrada dos blocos do oscilador é exemplificada na Figura 4.19. O bloco do oscilador cujo ganho originalmente é -10 é então construído com dois IAs em paralelo, um deles com ganho -10 e outro com ganho 1, resultando em um ganho equivalente a -9. O outro bloco é programado também com os IAs em paralelo com ganhos de 2 e -1, resultando em um bloco com ganho unitário. Se um do IAs em paralelo é desconectado devido a uma falha do tipo *stuck-open* o ganho do bloco será modificado, o que impacta diretamente na assinatura de tempo obtida.



**Figura 4.19:** Utilização de dois IAs como entrada dos blocos do oscilador

Para testar falhas do tipo *stuck-on* o oscilador é programado com sua configuração original (Figura 4.3), com apenas um IA utilizado em cada bloco. O IA vizinho ao amplificador cujo ganho é -10 é programado com um ganho igual a 9 e permanece

desconectado. Para o outro bloco do oscilador o IA que não está conectado ao laço do oscilador é programado com ganho unitário. Assim, se houver um *stuck-on* em uma das chaves dos IAs que deveriam permanecer desconectados, o ganho resultante do bloco será modificado, modificando assim a assinatura de tempo.

Nas configurações para o teste de falhas do tipo *stuck-open* em  $S_1$  e  $S_2$ , todos os IAs são conectados ao laço do oscilador. Logo, para testar este tipo de falha no oscilador apenas uma configuração de teste é necessária. Já para o teste de falhas do tipo *stuck-on*, apenas um dos IAs de cada CAB é testado em uma configuração. Logo, são necessárias 2 configurações para testar ocorrências de *stuck-on* nos dois CABs do oscilador. Para o teste da chave  $S_3$  dos dois CABs do oscilador mais duas configurações são necessárias. Como o procedimento precisa ser repetido para testar as chaves dos CABs que primeiramente foram utilizadas como ORA, 10 configurações são demandadas para testar todas as chaves do dispositivo segundo o modelo de falhas de colagem (*stuck-open* e *stuck-on*).

#### 4.2 AVALIAÇÃO DOS RESULTADOS

Nesta seção é feita uma avaliação dos resultados obtidos no teste dos CABs do FPPA ispPAC10. As duas estratégias aplicadas (BIST parcial e completo) são comparadas em termos de tempo de teste e cobertura de falhas.

A Tabela 3 sumariza o número de configurações necessárias para testar os diferentes componentes dos CABs do FPAA estudado, bem como a cobertura de falhas obtida em cada caso, conforme os dados apresentados nas seções anteriores.

**Tabela 3: Número de configurações e cobertura de falhas obtida (segundo o modelo adotado) para o método OBT com análise externa (OBT) e com análise interna de assinatura (ORA)**

Componente	Modelo de falhas	Falhas injetadas	Falhas cobertas		Não cobertas		Cobertura de falhas		Configurações	
			OBT	ORA	OBT	ORA	OBT	ORA	OBT	ORA
Chaves	Stuck_on/open	24	24	24	0	0	100%	100%	8	10
IAs	$K_{falha}=K\pm 1$	288	144	288	144	0	50%	100%	40	112
Capacitores	$\pm 20\%$	940	940	908	0	32	100%	96.6%	124	183
<b>Total</b>		1252	<b>1108</b>	<b>1220</b>	<b>144</b>	<b>32</b>	<b>88.5%</b>	<b>97.4%</b>	<b>172</b>	<b>305</b>

Comparando os resultados mostrados na Tabela 3 é possível perceber que a cobertura de falhas da estratégia envolvendo o ORA apresenta melhores resultados do que a estratégia

baseada no OBT com análise externa. Este último apresenta cobertura de falhas baixa para o teste dos blocos de ganho, enquanto o primeiro traz algumas dificuldades no teste de alguns valores de capacitores programáveis. Para o teste das chaves os dois esquemas resultam em uma cobertura de falhas de 100%.

Com o número total de configurações necessárias na aplicação destas estratégias, sabendo o tempo necessário para a programação do FPAA e considerando um tempo médio para avaliação do sinal ou para a análise de assinatura, é possível estimar o tempo total de teste. O tempo necessário para análise do sinal no método baseado em oscilação com análise externa é o tempo de *start up* e mais o tempo de um ciclo do sinal. O tempo relativo considerado neste trabalho para o estabelecimento da oscilação é de 6 ciclos completos (ARABI, KAMINSKA, 1999) e o período médio considerado nestas estimativas é 40 $\mu$ s. Já o tempo demandado pelo ORA para análise de assinatura é o próprio valor da assinatura de tempo. Baseado nos dados experimentais uma assinatura média de 8ms foi considerada nas estimativas.

De acordo com o manual do fabricante (LATTICE, 2000) o tempo de reconfiguração (apagar e escrever na memória) no pior caso é 200ms. Então é possível estimar o tempo de teste para as duas estratégias:

$$OBT: t_{test} = 172 \times [200ms + 6 \times 40\mu s] = 34.4s$$

$$ORA: t_{test} = 305 \times [200ms + 8ms] = 63.4s$$

O tempo de teste pode parecer alto para ambas estratégias, porém isto ocorre devido ao longo tempo de programação deste dispositivo em particular. De modo geral, o tempo de teste depende da complexidade do FPAA e do seu tempo de programação. O tempo de teste obtido com o ORA é maior, pois além de exigir um maior número de configurações há o tempo associado à própria assinatura. Contudo, se a análise de assinatura não for executada internamente é necessário um testador externo para avaliar a frequência e a amplitude do sinal oscilatório gerado.

Com a análise interna de assinatura um hardware externo muito simples se faz necessário, consistindo de um comparador e um contador. Assim, a assinatura de tempo pode ser codificada digitalmente e comparada com as assinaturas obtidas de um circuito sem falhas, armazenadas em uma memória, por exemplo. Desta maneira, o teste é executado por um

testador externo de muito baixo custo sem implicar no aumento de área em silício do dispositivo. Adicionalmente, este método pode ser de grande valia se considerado um ambiente do tipo SoC, pois recursos previamente existentes (um microcontrolador, por exemplo) podem ser usados para implementar tal testador, eliminando a necessidade de um equipamento externo. Outra possibilidade é mesclar as duas estratégias em um único procedimento de teste, de maneira a extrair as melhores características de cada uma, a fim de se obter um resultado otimizado.

## **5. TESTE DO FPAA AN221E04 DA ANADIGM COMPANY**

Devido a programabilidade estritamente funcional deste dispositivo a aplicação e o desenvolvimento de um método de teste também funcional se torna um tarefa menos difícil, se comparada com uma abordagem estrutural. Adicionalmente, sabendo que um único CAB pode implementar diferentes funções analógicas, cujas especificações são usualmente bastante amplas, existe a necessidade de utilizar vetores de teste simples e compactos (com amplo conteúdo harmônico). Uma alternativa para cumprir estes requerimentos é a utilização do método de teste funcional baseado na análise da resposta transiente (TRAM) que foi apresentado na seção 2.9.

Neste capítulo é descrita a aplicação do método TRAM orientado ao teste dos capacitores programáveis dos CABs do FPAA AN221E04. O procedimento de teste consiste em programar funções de transferência conhecidas utilizando o FPAA sob teste e aplicar estímulos a fim de se computar a resposta transiente. Funções de primeira e segunda ordem são consideradas. Inicialmente uma análise de sensibilidade é feita a fim de definir a função de transferência que maximize a observabilidade de falhas quando programada durante o teste. Os efeitos da transformação em frequência são também abordados. Em seguida o procedimento de teste é apresentado. Por fim, uma segunda análise de sensibilidade é feita, a fim de se identificar os mínimos desvios detectáveis nos capacitores programáveis dos CABs testados.

### **5.1 ANÁLISE DE SENSIBILIDADE DOS PARÂMETROS DA RESPOSTA TRANSIENTE**

Durante o projeto de circuitos analógicos um dos objetivos do projetista é obter uma topologia de circuito que desempenhe a função de interesse cumprindo as especificações de projeto e ao mesmo tempo obter funções cujos parâmetros de interesse possuam uma baixa sensibilidade a variações nos componentes do circuito. Esta característica é desejável justamente para tornar o circuito robusto a flutuações estatísticas no processo de produção ou a variações nas condições ambientais, o que pode modificar o valor de componentes e por consequência afetar o funcionamento do circuito.

Durante o teste se deseja detectar desvios nos componentes e identificar se estes se encontram dentro da faixa de tolerância prevista ou se o desvio é significativo a ponto de considerar o circuito falho. No teste funcional quanto maior a sensibilidade de um parâmetro

a um dado componente maior é capacidade de se observar falhas neste componente. No decorrer desta seção são realizadas análises de sensibilidade sobre os parâmetros da resposta transiente em relação à função de transferência de diferentes blocos funcionais. O objetivo destas análises é obter esquemas de configurações para os blocos analógicos programáveis do FPAA que resultem em uma sensibilidade relativamente alta, aumentando a observabilidade de falhas nos componentes.

Na seção 2.9 é possível encontrar as equações que modelam as funções de transferência de filtros passa baixa e passa alta de primeira ordem (Equações 6 e 7), e filtros passa-baixa, passa-alta e passa-banda de segunda ordem (Equações 8 a 10). Além destas funções serão estudadas neste capítulo funções de transferência de filtros rejeita faixa (BR – Band Reject), equalizador de atraso (DE – Delay Equalizer) e integrador puro (1ª ordem) cujas equações são dadas abaixo.

Integrador puro:

$$H_I = \frac{1}{s} \quad (11)$$

Filtro Rejeita-Faixa:

$$H_{BR}(s) = \frac{s^2 + \omega_0^2}{s^2 + \left(\frac{\omega_0}{Q}\right) \cdot s + \omega_0^2} \quad (12)$$

Equalizador de Atraso:

$$H_{DE}(s) = \frac{s^2 - \left(\frac{\omega_0}{Q}\right) \cdot s + \omega_0^2}{s^2 + \left(\frac{\omega_0}{Q}\right) \cdot s + \omega_0^2} \quad (13)$$

Conforme mencionado na seção 2.9, no método TRAM as funções que apresentam polinômios de ordem diferente de zero no numerador sofrem uma transformação de frequência, através da aplicação de um estímulo que corresponde a um *step* integrado uma ou

duas vezes (rampa ou parábola). Desta maneira o sistema pode ser visto como um filtro passa-baixa cujo sinal de entrada é um *step*.

As funções de transferência descritas pelas Equações 11 a 13 não resultam em filtros passa-baixa quando o sinal de entrada é integrado. A integração da função descrita pela Equação 11 (integrador) resulta em um duplo integrador e não em um filtro passa-baixa. Contudo o próprio bloco integrador pode ser utilizado no teste como função alvo, pois uma variação na constante de integração irá fazer com que a inclinação da rampa gerada pela integração do *step* também sofra uma modificação, permitindo assim que falhas sejam detectadas. Variações no ganho também podem ser detectadas utilizando um integrador puro.

Já os filtros rejeita-faixa e equalizador de atraso não podem ser transformados em passa-baixa através de integrações simples ou duplas. Segundo o autor do método TRAM a idéia principal da transformação de frequência é multiplicar a função de transferência do filtro em questão por uma outra função a fim de se obter a função de um filtro passa-baixa, não se restringindo apenas a multiplicação (em frequência) por integradores (CALVANO, 2005). Assim, para utilizarmos os blocos descritos pelas equações 12 e 13, a transformação em frequência consiste em aplicar um filtro ao sinal de entrada (*step*) cuja função de transferência é dada pelo inverso do polinômio do numerador do filtro que se deseja testar. Porém, a utilização destas funções torna a aplicação do teste mais custosa.

Sendo assim, a análise de sensibilidade dos parâmetros da resposta transiente considerada neste trabalho se reduz aos filtros passa-baixa de 1ª e 2ª ordem, pois todos blocos considerados podem ser transformados nestas funções através da multiplicação em frequência (convolução no tempo) por funções apropriadas.

O tempo de acomodação considerado para um resposta ao degrau de um filtro de 1ª ordem depende da constante de tempo ( $\tau$ ) do circuito, neste trabalho o tempo de acomodação é considerado como o tempo em que o valor da resposta atinge 98,2% do seu valor final (OGATA, 1982), ou seja:

$$T_s = 4\tau \quad (14)$$

Onde  $\tau$  depende de componentes físicos do circuito (geralmente resistores e capacitores). A frequência de corte de um filtro de primeira ordem tem uma relação inversamente proporcional ao parâmetro  $\tau$ , e por consequência ao parâmetro  $T_s$ :

$$\omega_0 = \frac{1}{\tau} \quad (15)$$

$$T_s = \frac{4}{\omega_0} \quad (16)$$

Do ponto de vista da sensibilidade não importa se o tempo de acomodação for considerado como 3, 4, ou 5 vezes o parâmetro  $\tau$ , pois a análise de sensibilidade é feita tomando-se as derivadas dos parâmetros envolvidos, e neste caso as constantes que multiplicam  $\tau$  não irão interferir nesta medida

Considerando como sinal de entrada um degrau unitário o valor final da amplitude do sinal de resposta ( $A$ ) será igual ao ganho do bloco funcional ( $G$ ), ou seja:

$$A = G \quad (17)$$

Esta relação também vale para os blocos de segunda ordem, pois o valor em estado estacionário também depende do ganho. A resposta ao degrau de uma rede de 2<sup>a</sup> ordem é mais complexa do que no caso das redes de 1<sup>a</sup> ordem, por isso há mais parâmetros que modelam o comportamento desta resposta. O tempo de acomodação ( $T_s$ ) depende de um parâmetro que vem da relação entre filtros elétricos e sistemas mecânicos: o fator de amortecimento ( $\xi$ ).

$$T_s = \frac{4}{\omega_n \xi} \quad (18)$$

O fator de amortecimento de sistemas mecânicos tem relação com o fator de qualidade ( $Q$ ) de filtros elétricos, pois ambos sistemas são modelados por equações diferenciais idênticas. Esta relação é descrita pela equação:

$$\xi = \frac{1}{2Q} \quad (19)$$

O tempo de pico ( $T_p$ ) depende da frequência amortecida ( $\omega_d$ ), cujo valor depende da frequência natural ( $\omega_n$ ) de oscilação (neste caso a frequência de corte), conforme a relação:

$$T_p = \frac{\pi}{\omega_d} \quad (20)$$

Onde

$$\omega_d = \omega_n \sqrt{1 - \xi^2} \quad (21)$$

O valor do pico de sobretensão (*overshoot*) também depende do fator de amortecimento:

$$Os = e^{-\pi \left( \frac{\xi}{\sqrt{1-\xi^2}} \right)} \quad (22)$$

Com as equações 14 a 22 é possível analisar a sensibilidade dos parâmetros da resposta transiente aos parâmetros da função de transferência do bloco considerado. Conforme já explicitado a análise será restrita a filtros passa baixa de 1ª e 2ª ordem.

A sensibilidade de um parâmetro da resposta transiente ( $P_i$ ) a variações em um parâmetro da função de transferência considerada ( $C_j$ ) é dada pela equação abaixo (DARYANANI, 1976):

$$S_{C_j}^{P_i} = \frac{C_j}{P_i} \cdot \frac{\partial P_i}{\partial C_j} \quad (23)$$

Utilizando a ferramenta computacional *Matlab* foi implementado um código que manipula as equações 14 a 23 e obtém as sensibilidades dos parâmetros da resposta ao degrau em relação aos parâmetros das funções de transferência de 1ª e 2ª ordem. Os códigos utilizados são mostrados no Apêndice A.

Os resultados das análises mostram que os parâmetros da resposta ao *step* relacionados com a frequência de corte do bloco sob teste possuem uma sensibilidade unitária (positiva ou negativa), conforme pode ser visto nas Tabelas 4 e 5, onde  $A_1$ ,  $T_{s1}$ ,  $G_1$  e  $f_{o1}$  são o valor final da amplitude no estado estacionário da resposta ao degrau, o tempo de acomodação, o ganho e a frequência de corte do bloco de primeira ordem respectivamente.  $A_2$ ,  $T_{s2}$ ,  $G_2$  e  $f_{o2}$  representam as mesmas grandezas quando considerado um bloco de segunda ordem.

**Tabela 4: Resultado da análise de sensibilidade em relação aos parâmetros “frequência de corte” e “ganho” para blocos de 1ª ordem**

$C_j$	$G_1$	$f_{o1}$
$P_i$	$A_1$	$T_{s1}$
$S_{C_j}^{P_i}$	1	-1

**Tabela 5: Resultado da análise de sensibilidade em relação aos parâmetros “frequência de corte” e “ganho” para blocos de 2ª ordem**

$C_j$	$G_2$	$f_{o2}$	
$P_i$	$A_2$	$T_{s2}$	$T_p$
$S_{C_j}^{P_i}$	1	-1	-1

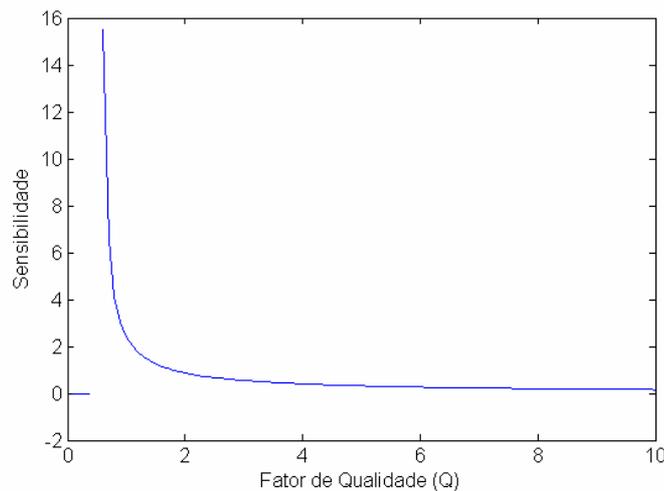
Uma sensibilidade unitária significa que uma variação percentual em um parâmetro da função de transferência ocasionará um desvio de igual magnitude nos parâmetros da resposta ao degrau. Exemplificando, se um parâmetro da função de transferência variar 10%, o parâmetro da resposta que apresenta uma sensibilidade unitária (positiva) varia também 10%, se a sensibilidade for unitária e negativa a variação é de  $-10\%$ .

A sensibilidade do tempo de estabilização ( $T_{s2}$ ) em relação ao fator de qualidade ( $Q$ ) do bloco de segunda ordem também apresenta um valor constante e igual a 1, porém a sensibilidade dos parâmetros  $T_p$  e  $O_s$  (tempo de pico e *overshoot*) apresentam relações mais complexas conforme as equações abaixo:

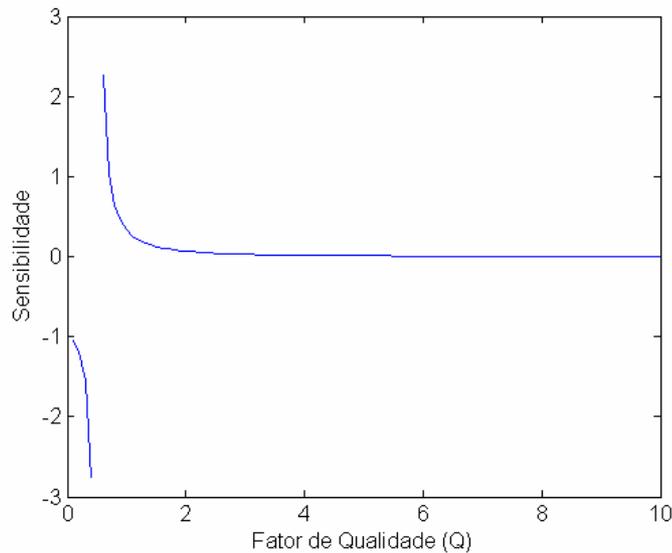
$$S_Q^{O_s} = \frac{4\pi Q^2}{(4Q^2 - 1)^{3/2}} \quad (24)$$

$$S_Q^{T_p} = \frac{1}{4Q^2 - 1} \quad (25)$$

As Figuras 5.1 e 5.2 mostram os gráficos que representam as relações dadas pelas equações 24 e 25 para valores de  $Q$  variando entre 0 e 10. Para valores maiores que 10 observa-se a mesma tendência em ambos os gráficos (decaimento assintótico até zero).



**Figura 5.1: Sensibilidade do parâmetro da resposta ao impulso “Overshoot” a variações no fator de qualidade de um filtro de segunda ordem**



**Figura 5.2: Sensibilidade do parâmetro da resposta ao impulso “Tempo de pico” a variações no fator de qualidade de um filtro de segunda ordem**

Nota-se nos dois gráficos uma descontinuidade em  $Q = 0,5$ . Para este valor de  $Q$  a sensibilidade do parâmetro “*Overshoot*” resulta em valores complexos com a parte real igual a zero enquanto o “Tempo de pico” obtido é negativo. Estas singularidades acontecem pois  $0,5$  é justamente o limite entre um sistema criticamente amortecido e um sistema sub-amortecido ou seja, para  $Q \leq 0,5$  não há oscilação na resposta transitória, logo, as grandezas físicas “*Overshoot*” e “Tempo de pico” deixam de existir (OGATA, 1976).

Por este motivo, considera-se neste trabalho apenas valores de  $Q$  maiores que  $0,5$ , pois quer-se detectar variações em parâmetros físicos e para isto estes parâmetros devem ser possíveis de ser visualizados. Vê-se também que quanto mais próximo de  $0,5$  for o valor de  $Q$  maior é a sensibilidade dos parâmetros considerados nas equações 20 e 21, porém, menor serão as magnitudes dos próprios parâmetros “*Os*” e “*Tp*”, o que também pode dificultar sua detecção. Em simulações realizadas com valor de  $Q$  igual ou inferior a  $0,5$  é possível visualizar que mesmo não havendo oscilação na resposta transitória há diferenças nestas respostas quando se varia  $Q$ , principalmente no parâmetro “Tempo de Acomodação”. Entretanto, estas variações não são maiores que as variações obtidas quando o valor de  $Q$  é próximo da unidade.

Como exemplo temos as sensibilidades teóricas obtidas em dois pontos para o parâmetro *Os*. Quando  $Q=0,6$  a sensibilidade do *Overshoot* em relação a  $Q$  é  $15,49$  e quando  $Q=0,707$ , a sensibilidade é igual a  $2\pi=6,28$ . Contudo para  $Q=0,6$  a amplitude da oscilação na resposta transitória é muito pequena o que a torna difícil de ser detectada.

Através dos dados obtidos na análise de sensibilidade e nas simulações é possível determinar uma faixa em que a detecção de variações no parâmetro  $Q$  (através da observação de  $Os$  e  $Tp$ ) seja facilitada. O intervalo considerado neste trabalho compreende os limites  $0,7 \leq Q \leq 1,5$  aproximadamente.

## 5.2 EFEITOS DA TRANSFORMAÇÃO EM FREQUÊNCIA

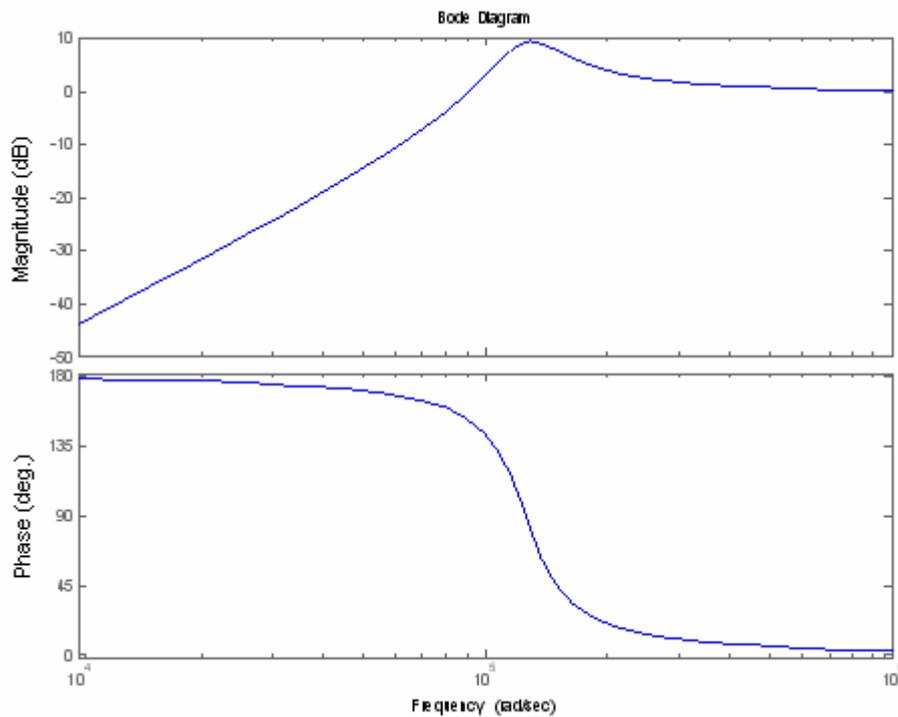
A transformação em frequência (multiplicação por  $1/s$  ou  $1/s^2$ ) dos filtros passa-alta e passa-banda resulta em uma rede passa-baixa. Porém, uma vez feita a transformação, o ganho do filtro passa-baixa resultante pode diferir do ganho do filtro considerado (passa-banda ou passa-alta). Considerando a função de transferência de um filtro passa-alta (Equação 5) ao proceder a transformação em frequência, multiplicando por  $1/s^2$ , obtém-se a seguinte função de transferência:

$$H_{HP'}(s) = \frac{G}{s^2 + \left(\frac{\omega_0}{Q}\right) \cdot s + \omega_0^2} \quad (26)$$

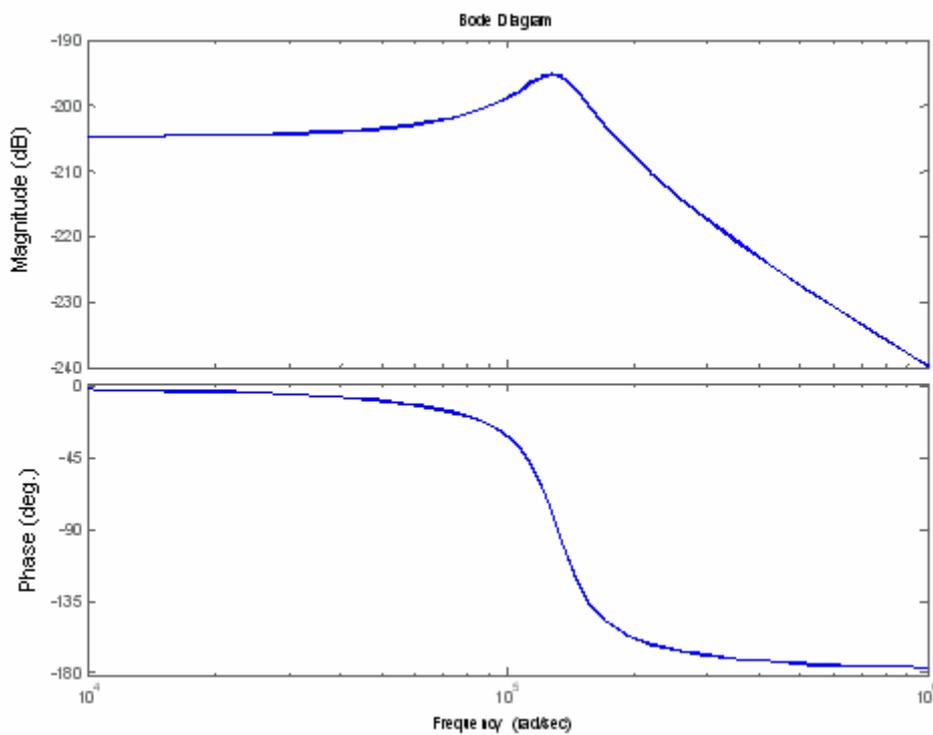
Observa-se que a equação modela um filtro passa baixa, porém, se compararmos a equação 22 com a equação 4 (função de transferência genérica de um filtro passa-baixa) é possível observar que nesta última tem-se o termo  $G\omega_0^2$  no numerador. Ao tomarmos o limite quando a frequência ( $s$ ) tende a zero obtemos o ganho na faixa de passagem de um filtro passa-baixa (DARYANANI, 1976), desta maneira é possível observar que o ganho do filtro resultante da transformação é:

$$G' = \frac{G}{\omega_0^2} \quad (27)$$

Isto significa que o ganho do filtro passa-baixa resultante será igual ao ganho do filtro passa-alta dividido pela frequência central do mesmo, podendo resultar em ganhos excessivamente baixos (altas atenuações). Este fato é ilustrado nas Figuras 5.3 e 5.4 que mostram respectivamente o diagrama de bode de um filtro passa-alta e do filtro passa-baixa resultante da transformação.



**Figura 5.3: Diagrama de bode de um filtro passa-alta cuja frequência de corte é aproximadamente 125Krad/s**



**Figura 5.4: Diagrama de bode de um filtro passa-baixa resultante da transformação em frequência do filtro passa-alta cujo diagrama de bode é mostrado na Figura 5.3**

Observa-se que o filtro passa-alta tem ganho unitário na faixa de passagem, porém o filtro passa-baixa resultante da transformação tem um ganho de  $-204\text{dB}$ , o que equivale a um ganho absoluto de  $1/125000^2$  que é exatamente o que se observa na equação 27. Uma vez que o ganho do filtro fica reduzido, a magnitude de sua resposta ao degrau também é reduzida na mesma escala, o que pode acarretar em dificuldades em observar o sinal, pois seria preciso equipamentos de medida com elevada precisão e sensibilidade. Teoricamente este problema pode ser solucionado aumentando a amplitude do estímulo de teste, ou seja, adicionando um bloco linear cujo ganho seja aproximadamente igual (em ordem de grandeza) a frequência central do filtro a ser testado. Desta maneira, no caso de um filtro passa-alta, o estímulo de entrada deve ser uma parábola multiplicada pela frequência de corte ao quadrado. Assim sendo, se considerarmos um *step* como estímulo de teste, este deve ser convoluído no tempo com um bloco cuja função de transferência (em frequência) é dada por:

$$H(s) = \frac{\omega_0^2}{s^2} \quad (28)$$

Logo, é possível concluir que a transformação em frequência integrando o estímulo de teste (*step*) deve ser acompanhada de um ganho a fim de não prejudicar a detecção da resposta transitória. Este ganho pode ser ajustado de maneira a obter a amplitude desejada da resposta transitória.

No caso de filtros passa-alta de segunda ordem, para obter um ganho unitário da resposta transitória em relação ao sinal de entrada, é preciso inserir um ganho igual a  $\omega_0^2$  no caminho do sinal. Para filtros passa-alta de primeira ordem este ganho deve ser igual a  $\omega_0$ . No caso de filtros passa-banda o ganho deve ser  $\omega_0 Q$ .

Na prática a transformação de frequência com adição de um bloco de ganho, nem sempre é factível. Dependendo dos valores das frequências de corte dos filtros envolvidos, o ganho que se deve adicionar a fim de se obter uma resposta ao *step* unitária pode ser muito elevado, superando as limitações do produto “ganho x faixa” dos componentes reais.

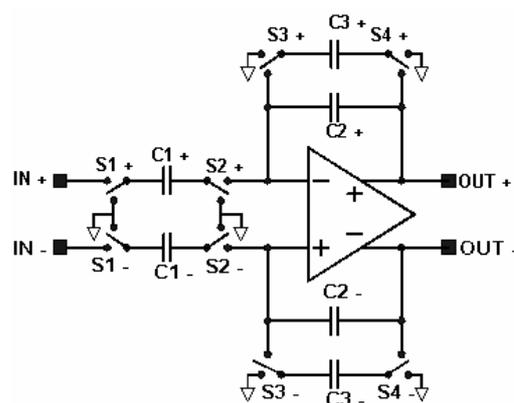
### 5.3 APLICAÇÃO DO TESTE

O FPAA considerado nesta etapa do trabalho como veículo de aplicação do teste é o componente AN221E04 da *Anadigm*, descrito na seção 3.3.2. O objetivo do procedimento descrito neste capítulo é testar os capacitores dos blocos programáveis (CABs) utilizando o método de análise de transiente.

Para facilitar a aplicação do teste, as funções escolhidas para serem programadas no FPAA são os filtros passa-baixa de primeira e segunda ordem. Desta maneira não há a necessidade de transformação em frequência (o que, conforme explicado anteriormente, nem sempre é realizável) e o estímulo a ser aplicado é um *step*. Tal fato facilita a geração do estímulo de teste.

É importante ressaltar que o objetivo não é testar a função de transferência programada e sim os componentes físicos que implementam esta função. A seção 5.5.1 mostra a relação entre os componentes dos CABs e os parâmetros das funções de transferência, bem como uma análise de sensibilidade, da qual se obtém o mínimo desvio detectável dos componentes de um CAB. Segundo o manual do fabricante, esta relação entre os parâmetros funcionais e os componentes de todos os blocos de segunda ordem é exatamente a mesma, o que resulta nos mesmos valores de sensibilidade para os blocos de segunda ordem. Por isso, os resultados obtidos se utilizados blocos passa-alta ou passa-banda seriam análogos aos resultados para o filtro passa-baixa, porém o estímulo de teste necessário seria mais complexo (parábola ou rampa). Finalmente, conforme visto anteriormente, dependendo da frequência central dos filtros passa-banda e passa-alta, o emprego destes blocos no teste seria impossibilitado do ponto de vista prático, devido ao alto ganho necessário para se obter uma resposta ao *step* unitária (facilitando assim sua observação).

A Figura 5.5 mostra o esquemático do filtro passa-baixa de 1<sup>a</sup> ordem considerado, já a Figura 5.6 mostra o esquemático do filtro passa-baixa de 2<sup>a</sup> ordem.



**Figura 5.5: Esquemático do filtro passa-baixa de 1<sup>a</sup> ordem utilizado**

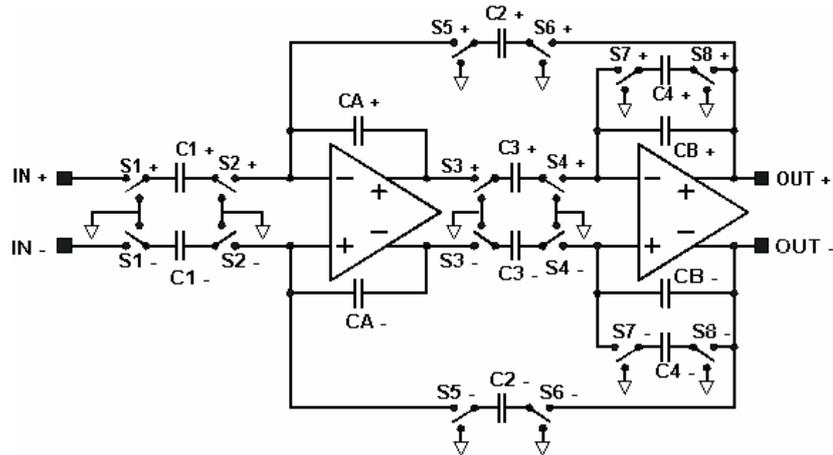


Figura 5.6: Esquemático do filtro passa-baixa de 2ª ordem utilizado

O CAB sob teste é então programado para implementar o filtro selecionado. O teste é baseado em redundância, ou seja, o filtro programado é duplicado e as saídas dos dois blocos são subtraídas, gerando um sinal de erro  $E(t)$ . O sinal  $E(t)$  é observado durante a aplicação do estímulo de teste e representa o desvio mútuo no valor das respostas ao *step* dos dois blocos. Este sinal de erro é então integrado, desta maneira, falhas que ocasionarem pequenos desvios nas respostas, porém por um período de tempo relativamente longo, são mais facilmente detectadas. Se a integral do sinal de erro sofrer um desvio maior que uma faixa de tolerância pré-determinada uma falha é detectada. Neste trabalho a faixa de tolerância considerada é  $\pm 30mV$ , pois, segundo o manual do fabricante (ANADIGM, 2003b), o *offset* de entrada dos CABs é  $15mV$  no pior caso e quando subtrai-se as saídas de dois CABs o *offset* resultante pode chegar a  $30mV$  (considerando também o pior caso).

O hardware adicional utilizado na detecção de desvios nas respostas dos dois blocos e o próprio gerador de estímulo de teste são concebidos utilizando os recursos programáveis disponíveis no FPAA. A Figura 5.7 ilustra o procedimento de teste tomando como exemplo a utilização de um filtro de segunda ordem. A Figura 5.8 mostra o diagrama em blocos do circuito de teste programado considerando a interface gráfica da ferramenta de programação *Anadigm Designer II*.

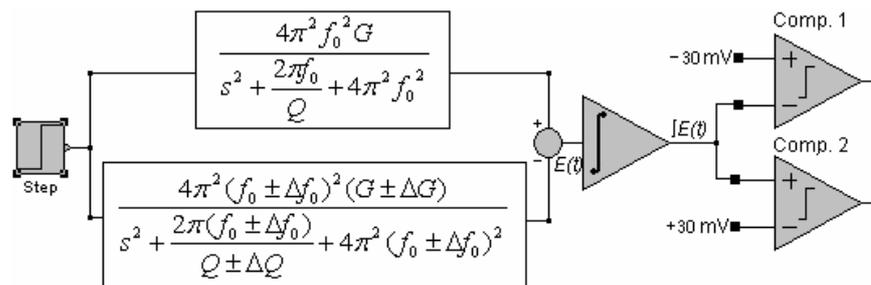


Figura 5.7: Diagrama em blocos da estrutura de teste considerando um filtro de segunda ordem

O comparador 1 indica que a integral do sinal de erro apresenta valor negativo e módulo maior que  $30mV$ , enquanto o comparador 2 satura quando este sinal é positivo e maior que  $30mV$ . A sensibilidade do circuito de detecção de erro pode ser ajustada modificando-se as constantes de integração do bloco integrador e a referência dos comparadores. Alternativamente, o sinal de erro integrado pode ser tomado como uma assinatura do circuito, pois, se nenhuma falha ocorrer, as respostas de ambos blocos tendem a ser iguais e o sinal de erro tende a ser zero e, por consequência, sua integral tende a zero também. Por outro lado, se uma falha afetar um dos blocos, o módulo do sinal de erro tende a ser maior que zero e o valor final da integração deste sinal tende a ser um nível DC não nulo. Porém, neste caso a probabilidade de *aliasing* não está descartada, pois o sinal de erro pode assumir valores positivos e negativos ao longo do tempo e, portanto, seu valor final integrado pode ser zero ou permanecer dentro da banda de tolerância.

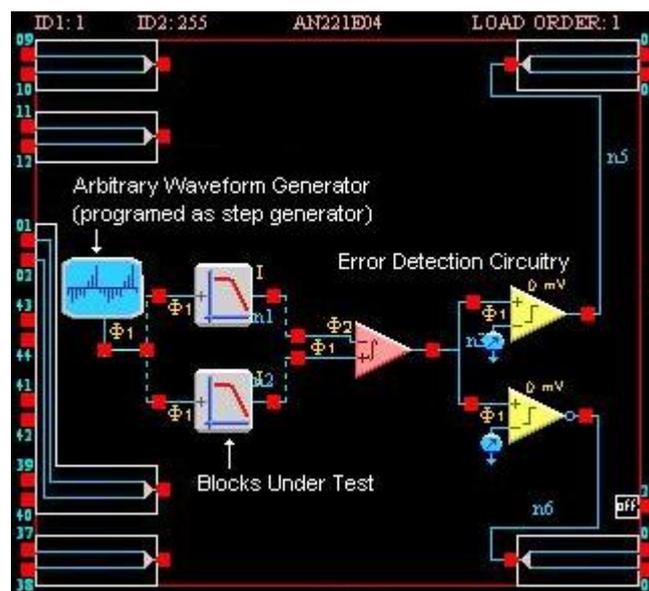


Figura 5.8: Área de trabalho do software *Anadigm Designer II* com a representação em blocos do circuito de teste programado

#### 5.4 INJEÇÃO DE FALHAS E SIMULAÇÃO

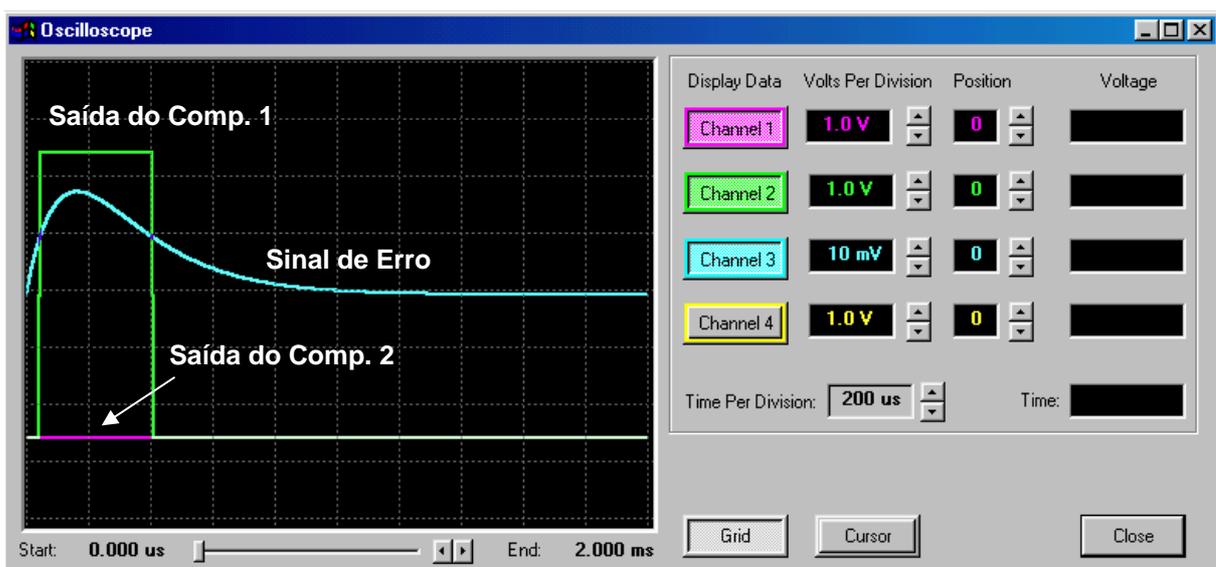
O modelo de falhas considerado nesta parte do trabalho é baseado em desvios dos parâmetros funcionais do bloco sob teste. Os parâmetros de alto nível da função de transferência são modificados para simular o efeito de uma falha. Além da frequência central

( $\omega_0=2\pi f_0$ ), foram injetadas falhas nos parâmetros fator de qualidade ( $Q$ ) e ganho ( $G$ ) das funções de transferência utilizadas no teste .

As falhas foram injetadas através do software de programação do dispositivo, mais uma vez, utilizando os recursos programáveis do FPAA. Desta maneira é possível simular o comportamento do circuito na presença de falhas configurando o circuito para que os parâmetros das funções de transferência apresentem desvios em relação ao valor nominal considerado.

Inicialmente foram realizadas simulações sem considerar o integrador no circuito detector de erro, apenas realizando a diferença dos sinais e comparando-os com a janela de referência considerada. Os resultados das simulações com o software *Anadigm Designer II* mostraram que variações maiores que 5% em módulo nos parâmetros  $Q$  e  $f_0$  são detectados pelo circuito analisador de resposta. As Figuras 5.9, 5.10 e 5.11 mostram, respectivamente, os sinais de erro (não integrados) obtidos para variações nos parâmetros frequência de corte (para os blocos de 1ª e 2ª ordem) e fator de qualidade (2ª ordem).

A inclusão do bloco integrador melhora a observabilidade do sinal de erro. Variações em torno de 3% nos parâmetros  $Q$  e  $f_0$  são detectados pelo circuito analisador de resposta quando realizada a integração do sinal. Variações no ganho também são facilmente detectadas uma vez que a integração da diferença do valor final das respostas dos blocos utilizados no teste resultará em uma rampa que certamente causará a saturação de um dos dois comparadores.



**Figura 5.9: Sinal de Erro não integrado e saída dos comparadores para um desvio injetado de +5% na frequência de corte do bloco de 1ª ordem**

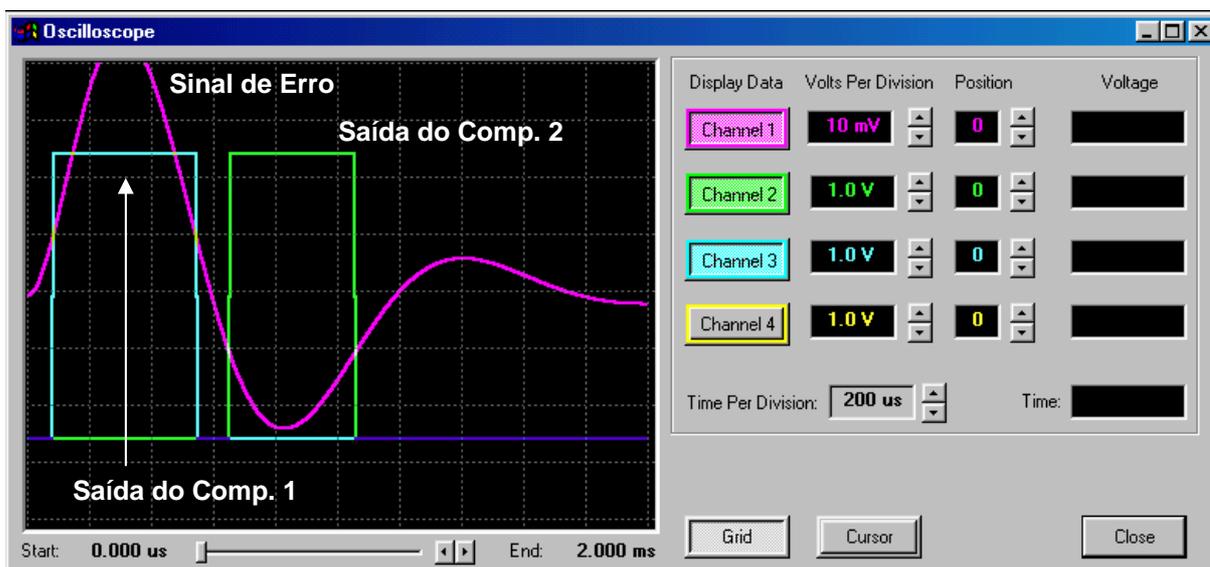


Figura 5.10: Sinal de Erro não integrado e saída dos comparadores para um desvio injetado de +5% na frequência de corte do bloco de 2<sup>a</sup> ordem

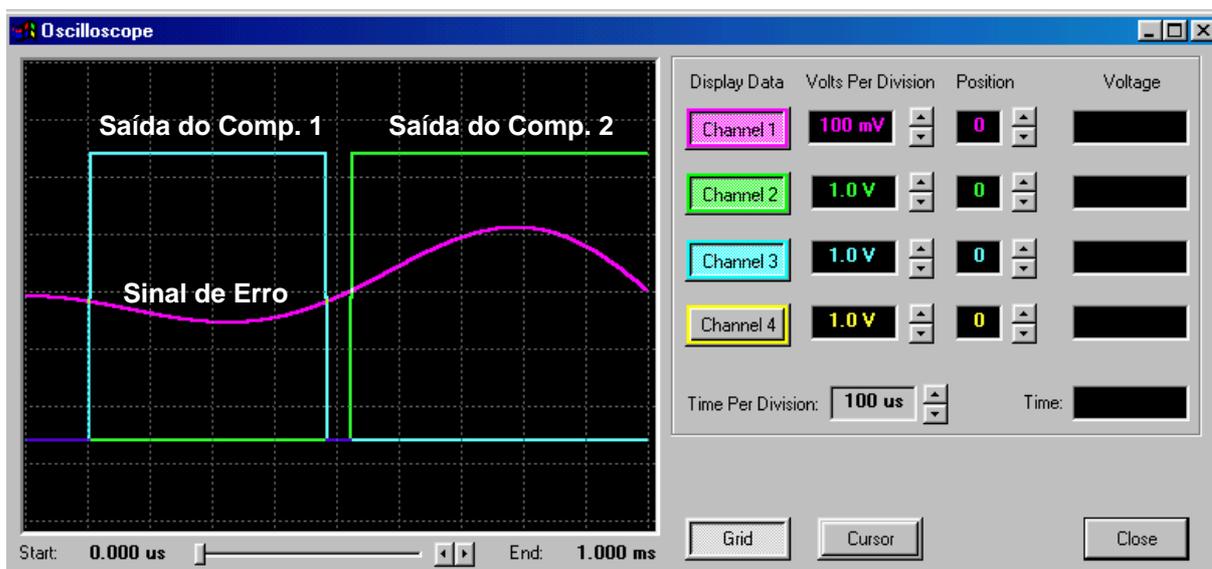


Figura 5.11: Sinal de Erro não integrado e saída dos comparadores para um desvio injetado de +5% no fator de qualidade do bloco de 2<sup>a</sup> ordem

## 5.5 RESULTADOS EXPERIMENTAIS

### 5.5.1 Análise de Sensibilidade Dos Parâmetros Funcionais do Circuito

A relação matemática entre os parâmetros de alto nível das funções de transferência e os componentes físicos do FPAA AN221E04 pode ser encontrada no manual dos módulos de IP

(*Intellectual Property*) disponíveis na biblioteca de programação do componente (ANADIGM, 2002). As equações 29 e 30 mostram, respectivamente, as expressões para o ganho ( $G_1$ ) e frequência de corte ( $f_{01}$ ) considerando o bloco de 1<sup>a</sup> ordem. Analogamente, as relações entre os parâmetros  $G_2$ ,  $f_{02}$  e  $Q$  com os componentes do circuito são explicitadas nas equações 31 a 33. Os componentes mostrados nestas equações podem ser visualizados nas Figuras 5.5 e 5.6.

Apesar deste dispositivo utilizar a tecnologia de capacitores chaveados, é possível realizar as análises do circuito no domínio “s” quando a frequência de chaveamento dos capacitores for suficientemente maior do que a frequência do sinal a ser processado pelo FPAA.

$$G_1 = \frac{C_1}{C_3} \quad (29)$$

$$f_{01} = \frac{f_c}{\pi} \cdot \frac{C_3}{(2C_2 + C_3)} \quad (30)$$

$$G_2 = \frac{C_1}{C_2} \quad (31)$$

$$f_{02} = \frac{f_c}{2\pi} \cdot \sqrt{\frac{C_2 \cdot C_3}{C_A \cdot C_B}} \quad (32)$$

$$Q = \frac{C_B}{C_4} \cdot \sqrt{\frac{C_2 \cdot C_3}{C_A \cdot C_B}} \quad (33)$$

Nas Equações 30 e 31,  $f_c$  é a frequência de chaveamento (dos capacitores) utilizada pelos módulos programados no FPAA.

Nas equações 29 a 33 os valores dos capacitores dos ramos positivo e negativo (dos circuitos totalmente diferenciais dos filtros) são considerados iguais, i. e.,  $C_{j+} = C_{j-}$ . Entretanto, neste trabalho assume-se que uma falha pode ocorrer em um ramo isolado, desbalanceando o circuito. Por esta razão as funções de transferência dos filtros foram analisadas considerando  $C_{j+}$  diferente de  $C_{j-}$ . As novas relações entre os componentes e os parâmetros do circuito, considerando tal fato, são mostradas nas seguintes equações:

$$G_{1FD} = \frac{\frac{C_{1+}}{C_{3+}} + \frac{C_{1-}}{C_{3-}}}{2} \quad (34)$$

$$G_{2FD} = \frac{\frac{C_{1+}}{C_{2+}} + \frac{C_{1-}}{C_{2-}}}{2} \quad (35)$$

$$f_{01FD} = \frac{f_C}{\pi} \cdot \sqrt{\frac{C_{3+}}{(2C_{2+} + C_{3+})} \cdot \frac{C_{3-}}{(2C_{2-} + C_{3-})}} \quad (36)$$

$$f_{02FD} = \frac{f_C}{2\pi} \cdot \sqrt{\sqrt{\frac{C_{2+} \cdot C_{3+}}{C_{A+} \cdot C_{B+}}} \cdot \sqrt{\frac{C_{2-} \cdot C_{3-}}{C_{A-} \cdot C_{B-}}}} \quad (37)$$

$$Q_{FD} = \frac{\frac{C_{B+}}{C_{4+}} \cdot \sqrt{\frac{C_{2+} \cdot C_{3+}}{C_{A+} \cdot C_{B+}}} + \frac{C_{B-}}{C_{4-}} \cdot \sqrt{\frac{C_{2-} \cdot C_{3-}}{C_{A-} \cdot C_{B-}}}}{2} \quad (38)$$

Nas equações 34 a 38 os índices “+” e “-” indicam o ramo do circuito no qual o capacitor considerado se encontra (positivo ou negativo) e o índice “FD” significa (*Fully Differential* do inglês: totalmente diferencial).

A estrutura de teste mostrada nas Figuras 5.7 e 5.8 é então programada na placa de desenvolvimento do FPAA AN221E04. As mesmas falhas consideradas no processo de simulação são injetadas no circuito, modificando-se os valores dos parâmetros programados de um dos CABs sob teste. A injeção de falhas foi realizada para diversos valores de  $f_0$ ,  $Q$  e  $G$ , ao longo de suas escalas de valores programáveis.

Conforme previsto nas simulações, os resultados obtidos mostram que variações de  $\pm 3\%$  nos parâmetros  $Q$  e  $f_0$  das funções de transferência são detectadas pelo circuito de observação do sinal de erro. A Figura 5.12 mostra as formas de onda adquiridas no hardware (sinal de erro integrado, respostas ao *step* com e sem falhas e a saída do comparador 1) considerando uma variação de  $+20\%$  na frequência de corte de um dos blocos sob teste (de  $10KHz$  para  $12KHz$ ).

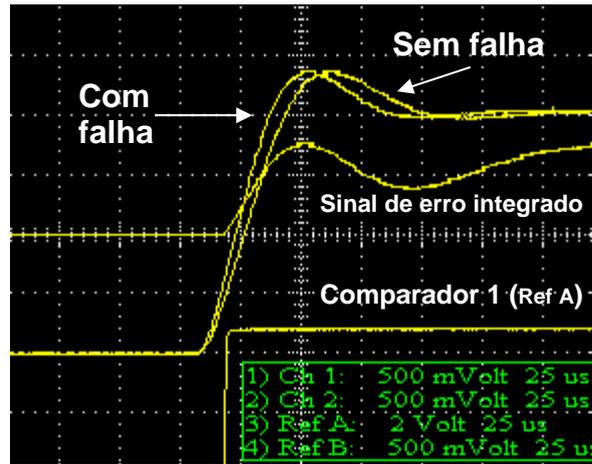


Figura 5.12: Formas de onda adquiridas durante um dos ensaios de injeção de falhas

Utilizando as equações 34 a 38 uma análise de sensibilidade foi realizada a fim de se investigar de que maneira e intensidade uma variação nos capacitores de um CAB afetam os parâmetros das funções de transferência dos blocos programados. A sensibilidade de cada parâmetro da função de transferência ( $P_i$ ) em relação a variações nos componentes dos CABs ( $C_j$ ) é então calculada, de acordo com a equação 39.

$$S_{C_j}^{P_i} = \frac{C_j}{P_i} \cdot \frac{\partial P_i}{\partial C_j} \quad (39)$$

Onde,

$$P_i = \{G_{1FD}, G_{2FD}, f_{01FD}, f_{02FD}, Q_{FD}\}$$

$$C_j = \{C_{1+}, C_{2+}, C_{3+}, C_{4+}, C_{A+}, C_{B+}, f_c, C_{1-}, C_{2-}, C_{3-}, C_{4-}, C_{A-}, C_{B-}\}$$

As equações 34 a 38 dependem de um grande número de variáveis e apresentam termos envolvendo divisões e multiplicações por raiz quadrada. Por esta razão, em alguns casos a análise de sensibilidade resulta em equações com muitos termos. Para manter a concisão e facilitar a compreensão desta etapa do trabalho, após o cálculo das equações de sensibilidade, é considerado o caso limite em que os componentes do ramo positivo e negativo tendem a ser iguais. O Apêndice B mostra o código desenvolvido com a ferramenta *Matlab* para cálculo das sensibilidades.

As Tabelas 6 e 7 mostram os melhores valores de sensibilidade (do ponto de vista do teste) para cada componente dos blocos de 1ª e 2ª ordem respectivamente, bem como os parâmetros das funções de transferência relacionados.

**Tabela 6: Resultados da análise de sensibilidade para o filtro de 1ª ordem**

$C_j$	$C_1$	$C_2$	$C_3$	$f_c$
$P_i$	$G_{1FD}$	$f_{01FD}$	$G_{1FD}$	$f_{01FD}$
$S_{C_j}^{P_i}$	<b>0,5</b>	$\frac{-C_2}{2C_2 + C_3}$	<b>-0,5</b>	<b>1</b>

**Tabela 7: Resultados da análise de sensibilidade para o filtro de 2ª ordem**

$C_j$	$C_{1+/-}$	$C_{2+/-}$	$C_{3+/-}$	$C_{4+/-}$	$C_{A+/-}$	$C_{B+/-}$	$f_c$
$P_i$	$G_{2FD}$	$G_{2FD}$	$f_{02FD},$ $Q_{FD}$	$Q_{FD}$	$f_{02FD},$ $Q_{FD}$	$f_{02FD},$	$f_{02FD}$
$S_{C_j}^{P_i}$	<b>0,5</b>	<b>-0,5</b>	<b>0,25</b>	<b>-0,5</b>	<b>-0,25</b>	<b>-0,25</b>	<b>1</b>

É possível notar através da Tabela 6 que o melhor resultado obtido para a sensibilidade do parâmetro  $f_{01}$  a variações em  $C_2$  não é um valor constante e depende também de  $C_3$ . Contudo os capacitores  $C_2$  e  $C_3$  podem ser programados de maneira que a sensibilidade seja maximizada, atingindo um valor muito próximo a  $-0,5$ .

De acordo com estes resultados e considerando o mínimo desvio detectável pelo circuito de avaliação de resposta como sendo  $\pm 3\%$ , a mínima variação possível de se detectar nos capacitores pode ser estimada, conforme os dados mostrados na Tabela 8.

**Tabela 8: Variações mínimas detectáveis nos valores dos capacitores dos CABs**

Capacitor	Bloco de 1ª ordem	Bloco de 2ª ordem
$C_{1+/-}$	<b>6%</b>	<b>6%</b>
$C_{2+/-}$	<b>6%</b>	<b>6%</b>
$C_{3+/-}$	<b>6%</b>	<b>12%</b>
$C_{4+/-}$	-	<b>6%</b>
$C_{A+/-}$	-	<b>12%</b>
$C_{B+/-}$	-	<b>12%</b>

Da Tabela 8 conclui-se que utilizando o método de análise de transiente para este modelo específico de FPAA é possível detectar desvios superiores a 6% (em módulo) quando utilizado o bloco de primeira ordem para aplicar o teste. Se o filtro de segunda ordem é

utilizado, variações maiores que 6% e 12% (dependendo do capacitor considerado) podem ser detectadas. Além disto, variações maiores que 3% na frequência de chaveamento dos capacitores ( $f_c$ ) são também detectadas.

Os resultados indicam que o filtro de 1ª ordem apresenta uma maior sensibilidade a variações nos componentes do CAB utilizado em sua programação. Por outro lado, o filtro de 2ª ordem é construído utilizando-se mais componentes de um único CAB, o que significa que um número maior de componentes é testado ao mesmo tempo, diminuindo o tempo de teste. Logo, dependendo do parâmetro de teste que se deseja otimizar (cobertura de falhas ou tempo de aplicação do teste) pode-se escolher o bloco mais adequado ou mesmo implementar um procedimento de teste misto.

### 5.5.2 Estimativa da Cobertura de Falhas e Tempo de Teste

No esquema de teste proposto dois CABs do FPAA são testados em paralelo enquanto os dois CABs restantes são utilizados na implementação do circuito analisador de resposta. Desta maneira, duas configurações estruturais são necessárias para testar os 4 CABs do FPAA, pois necessita-se repetir o procedimento para contemplar o teste dos CABs que no primeiro momento serviram na implementação do circuito detector de erro.

Na figura 5.6 nota-se que são necessários seis capacitores distintos na implementação do filtro de segunda ordem, embora cada CAB possua oito bancos de capacitores. Por esta razão, o procedimento global de teste deve ser repetido, modificando os bancos de capacitores utilizados de maneira que todos os bancos sejam então testados. Assim, o número de configurações estruturais necessárias é multiplicada por dois, totalizando 4 configurações.

Cada capacitor programável pode assumir 255 valores diferentes de capacitância. Logo, com 4 configurações estruturais necessárias para cobrir todos os bancos de capacitores do dispositivo, o número de configurações de teste é estimada em 1020 ( $255 \times 4$ ).

Considerando também que cada capacitor programável é configurado utilizando apenas um capacitor do banco por vez, é possível estender os resultados da Tabela 8 para cada capacitor individual dos bancos. Isto significa que, por exemplo, uma variação de 12% em um dos capacitores do banco representará um desvio de 12% no capacitor programado.

Já o bloco de 1ª ordem é composto por três capacitores, logo, três configurações estruturais são necessárias para cobrir os 8 bancos. Adicionalmente, apenas um dos dois *OPAMPs* disponíveis no CAB é utilizado na implementação do circuito. Uma vez que falhas

nos capacitores podem depender de sua conexão com os amplificadores operacionais, os dois *OPAMPs* necessitam ser utilizados. Desta maneira, seis configurações estruturais são necessárias para cobrir dois CABs. Para os outros dois CABs mais seis configurações são necessárias. Assim, 12 configurações estruturais são necessárias quando utilizado o filtro de primeira ordem na aplicação do teste. Portanto, o número de configurações de teste é estimado em 3060 ( $255 \times 12$ ).

Considerando um modelo de falhas paramétricas consistindo em desvios de  $\pm 15\%$  nos capacitores do CAB sob teste é possível obter uma cobertura de falhas de 100%, uma vez que o mínimo desvio detectável é 12% no pior caso (Tabela 8). Este modelo de falhas representa uma boa aproximação, pois a precisão absoluta típica no processo de fabricação de capacitores MOS é aproximadamente 10%, apesar de se obter precisões relativas (“*matching*”) entre capacitores de um mesmo CI tão boas quanto 0,1% (ALLEN; HOLBERG, 1987).

Com o número total de configurações estimadas para este procedimento de teste, sabendo o tempo necessário para efetuar a programação do FPAA e considerando um tempo mínimo para avaliação da resposta transiente é possível estimar o tempo total de teste.

Se, em cada configuração de teste, os capacitores programáveis forem configurados com um único capacitor do banco por vez e todos os capacitores do circuito programados com o mesmo valor, os parâmetros do bloco considerado serão constantes ( $G_1=G_2=1$ ,  $f_{01}=f_{02}=f_c/2\pi$ , e  $Q=1$ .) como pode ser visto nas equações 29 a 33. Isto significa que os parâmetros da resposta ao *step* podem ser mantidos constantes durante todo procedimento de teste. Desta maneira, o tempo de acomodação, que influencia diretamente no tempo de teste, pode ser programado para apresentar um valor que contribua para a diminuição deste tempo. O tempo de acomodação considerado na estimativa do tempo de teste é  $25\mu s$ . Adicionalmente, o valor de  $Q$  pode ser programado de maneira a obter-se um valor desejável da sensibilidade dos parâmetros “*Overshoot*” e “Tempo de Pico”, conforme os resultados da seção 5.1.

De acordo com o manual do fabricante, a maneira mais rápida de se programar o FPAA AN221E04 é através de um microcontrolador dedicado. O tempo necessário para a programação no “modo microcontrolador” pode ser estimado em  $150\mu s$ . Portanto, os tempos totais de teste utilizando os filtros de 2ª e 1ª ordem podem ser estimados como:

$$t_{test\_2nd} = 1020TCs \cdot (150 + 25)\mu s = 178.5ms$$

$$t_{test\_1st} = 3060TCs \cdot (150 + 25)\mu s = 535.5ms$$

Onde TCs significa *Test Configurations* (Configurações de Teste).

## 6. CONCLUSÕES

Desde o surgimento dos FPAA's poucos trabalhos relacionados ao teste deste tipo de dispositivo foram desenvolvidos, sendo este trabalho um dos primeiros a considerar este problema. Devido a este caráter pioneiro diversos aspectos foram abordados, abrindo um leque de possibilidades a serem exploradas no teste destes componentes.

As estratégias de teste desenvolvidas ao longo do trabalho foram validadas considerando dois FPAA's de características distintas. A primeira estratégia desenvolvida trata-se de um esquema de teste estrutural que utiliza o teste baseado em oscilação para detectar falhas nos componentes dos blocos analógicos programáveis de FPAA's. Esta estratégia foi aplicada a um FPAA de tempo contínuo da família ispPAC da *Lattice Semiconductors*, considerando um modelo falhas estrutural. Este modelo consiste em desvios paramétricos nos valores dos capacitores e ganhos dos IAs, bem como falhas do tipo *stuck-at* em algumas chaves programáveis do FPAA.

O teste baseado em oscilação associado à análise externa apresentou uma cobertura de falhas de 100% para os capacitores e para as chaves, segundo o modelo de falhas adotado. Já para os blocos de ganho, uma baixa sensibilidade dos parâmetros de teste, em determinadas faixas de valores programáveis, impôs uma baixa cobertura de falhas (50%). Entretanto, a cobertura de falhas total obtida com a aplicação deste método foi de 88,5%. Uma significativa melhoria na cobertura de falhas foi obtida com a utilização de um analisador interno de assinatura (ORA) baseado em um duplo integrador. Esta melhoria se deve ao aumento da observabilidade de falhas nos blocos de ganho, pois o ORA é sensível ao tempo de estabelecimento (*start up*) da oscilação, parâmetro que depende diretamente do ganho total de laço do oscilador.

A cobertura de falhas obtida com adição do analisador de assinatura foi de 97,4%. Contudo, o tempo de teste demandado quando utilizado o ORA é aproximadamente 85% maior do que no caso da análise externa. Isto ocorre devido ao maior número de configurações de teste necessárias na implementação do ORA e à própria assinatura obtida, que é uma medida de tempo, e, por isso, acaba interferindo no tempo de teste. Os tempos de teste obtidos foram altos, sendo 34,4 segundos para o método OBT "puro" e 63,4 segundos com a adição do ORA. Contudo ainda há espaços para melhorias neste parâmetro, pois é possível derivar configurações nas quais as chaves programáveis, blocos de ganho e capacitores sejam testados ao mesmo tempo. O tempo de programação do FPAA também é

um parâmetro determinante no tempo de teste. Tecnologias nas quais o tempo de teste é menor necessitam de menos tempo para cobrir todas as configurações de teste. O conhecimento de detalhes específicos da arquitetura do circuito (como é o caso do teste em produção) também pode acarretar em diminuição do tempo de teste.

A outra estratégia desenvolvida neste trabalho utiliza um método de teste funcional baseado em análise de transiente, conhecido como TRAM. O método TRAM é então associado a um esquema de teste baseado em redundância. O bloco sob teste é duplicado e as saídas dos dois blocos são comparadas, visando detectar desvios mútuos na resposta transiente. Esta estratégia foi aplicada ao teste do FPAA AN221E04. O teste teve seu foco nos blocos analógicos programáveis deste dispositivo. Duas diferentes funções de transferência foram programadas no FPAA sob teste, e os resultados obtidos foram comparados em termos de cobertura de falhas e tempo de aplicação do teste. Os parâmetros da função de primeira ordem apresentam uma maior sensibilidade a variações nos capacitores dos CABs, facilitando a detecção de pequenos desvios paramétricos. Por outro lado o bloco de segunda ordem compreende um número maior de capacitores, implicando em uma maior cobertura de componentes de um CAB, o que reduz o número de configurações necessárias e conseqüentemente o tempo de teste.

O gerador de estímulo de teste (*step*) e o circuito de análise de resposta foram construídos com os recursos programáveis disponíveis no FPAA, simplificando a geração e a análise da resposta de teste sem acarretar em *overhead* (aumento) de área em silício ocupada fisicamente pelo dispositivo. A saída do circuito analisador de resposta de teste é um par de sinais digitais (saída dos comparadores), facilitado a integração com um sistema digital externo.

Os resultados obtidos foram satisfatórios. Desvios mínimos paramétricos de 6% a 12% nos capacitores dos CABs podem ser detectados (dependendo do bloco utilizado no teste). Adicionalmente, os 32 bancos de capacitores (8160 capacitores) do FPAA AN221E04 podem ser testados em tempos estimados em 187.5ms e 535.5ms (também dependendo da função utilizada na aplicação do teste).

O método TRAM mostrou-se adequado para o teste funcional de FPAA's principalmente quando detalhes arquiteturais do circuito alvo são desconhecidos. Obviamente, se o objetivo for o teste em produção, os detalhes do circuito são de fato conhecidos e técnicas estruturais podem ser aplicadas. Contudo, esta metodologia funcional pode ser utilizada como um auto-teste em campo, sendo periodicamente aplicada para garantir o correto funcionamento do

circuito. Outra possibilidade é mesclar esta estratégia com métodos estruturais com o objetivo de aumentar a cobertura de falhas ou reduzir o tempo de teste.

Como proposta de trabalhos futuros fica a utilização de modelos de falhas mais realistas (considerando falhas também nos OPAMPs), porém, para isso existe a necessidade do conhecimento dos detalhes do circuito.

Um trabalho já em estágio inicial visa estudar o efeito dos chamados SEUs (Single Event Upsets) (MESSENGER, 1992) nos FPAAs cuja memória de programação é do tipo SRAM, bem como desenvolver técnicas de tolerância a falhas para os projetos que utilizam este tipo de dispositivo. Tais técnicas podem encontrar suporte na programabilidade dos FPAAs, o que pode permitir a implementação de circuitos auto-calibráveis ou auto-recuperáveis (*self recovering*).

As estratégias de teste desenvolvidas neste trabalho foram aplicadas a modelos específicos de FPAAs. Porém, pode-se estendê-las a componentes de diferentes fabricantes, uma vez que os blocos funcionais utilizados em ambas abordagens são simples (integradores e filtros de 1<sup>a</sup> e 2<sup>a</sup> ordem) e, por isso, estima-se que possam ser programados em outros modelos de FPAAs.

## REFERÊNCIAS

ALLEN, P.; HOLBERG, D. **CMOS Analog Circuit Design**. New York, USA: Holt-Rinehart and Winston, 1987.

ANADIGM. Anadigm Designer IP Module Manual. **Datasheet**, 2002. Disponível em [www.anadigm.com](http://www.anadigm.com). Acesso em: 8 mar. 2006.

ANADIGM. AN10E40 Field Programmable Analog Array. **DataSheet**, 2003. Disponível em [www.anadigm.com](http://www.anadigm.com). Acesso em: 8 mar. 2006.

ANADIGM. AN221E04 Field Programmable Analog Array. **DataSheet**, 2003b. Disponível em [www.anadigm.com](http://www.anadigm.com). Acesso em: 8 mar. 2006.

ANDRADE JR. A. Q. *et al.* Built-in Self-Test of Global Interconnects of Field Programmable Analog Arrays. **Microelectronics Journal**, Kidlington, UK: Elsevier, v. 36, n.12, p. 1112 – 1123, 2005.

ARABI, K.; KAMINSKA, B. Oscillation Test Strategy for Analog and Mixed-Signal Integrated Circuits. In: VLSI TEST SYMPOSIUM, 14., 1996, Princeton, USA. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 1996, p. 476-482.

ARABI, K.; KAMINSKA, B. Testing Analog and Mixed-Signal Integrated Circuits Using Oscillation-Test Method. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, USA: IEEE, v. 16, n.7, p. 745-753, july 1997.

ARABI, K.; KAMINSKA, B. Efficient and Accurate Testing of Analog-to-Digital Converters Using Oscillation-Test Method. In: EUROPEAN DESIGN AND TEST CONFERENCE, 1997, Paris, France. **Proceedings...** [S. l.: S. n.], mar. 1997b, p.348 – 352.

ARABI *et al.* Digital Oscillation-Test Method for Delay and Stuck-at Fault Testing of Digital Circuits. In: INTERNATIONAL TEST CONFERENCE, 1998, Washington DC, USA. **Proceedings....** Washington DC, USA: International Test Conference Press, oct. 1998, p. 91 – 100.

ARABI, K.; KAMINSKA, B. Design for Testability of Embedded Integrated Operational Amplifiers. **IEEE Journal of Solid-State Circuits**. New York, USA: IEEE, v. 33, n. 4, p. 573- 581, apr. 1998b.

ARABI, K.; KAMINSKA, B. Oscillation-Test Methodology for Low-Cost Testing of Active Analog Filters. **IEEE Transactions on Instrumentation and Measurement**, New York, USA: IEEE, v. 48, n. 4, p. 798 – 806, aug. 1999.

- BALEN, T. R. *et al.* Testing the Configurable Analog Blocks of Field Programmable Analog Arrays. In: INTERNATIONAL TEST CONFERENCE, 2004, Charlotte, USA. **Proceedings...** Washington DC, USA: International Test Conference Press, oct. 2004, p. 893 – 902.
- BALEN, T. R. *et al.* Applying the Oscillation Test Strategy to FPAA's Configurable Analog Blocks, **Journal of Electronic Testing: theory and applications**, New York, USA: Springer Science + Business Media, v. 21, n. 2, p. 135-146, 2005.
- BALEN, T. R. *et al.* Functional Test of Field Programmable Analog Arrays. In: IEEE VLSI Test Symposium, 24., 2006, Napa Valley, USA. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, may, 2006, p. 326 – 333.
- BALEN, T. R.; LUBASZEWSKI, M. S.; RENOVELL, M. Study of Single Event Upset effects in SRAM-Based Field Programmable Analog Arrays. In: IEEE LATIN-AMERICAN TEST WORKSHOP, 7., 2006, Buenos Aires, Argentina. **Digest of Papers...**, [S. l.: S. n.], 2006b, p. 115-119.
- BELL, I. M.; SPINKS, S. J. Evaluation and Comparison of Structural Test Methodologies for Analogue and Mixed Signal Circuits. In: IEEE INTERNATIONAL CONFERENCE ON ELECTRONICS, CIRCUITS AND SYSTEMS, 1998, [S. l.]. **Proceedings...**[S. l.: S. n.], v. 2, sept. 1998, p. 433 – 436.
- BRATT, A.; MACBETH, I. DPAD2: a field programmable analog array. **Analog Integrated Circuits and Signal Processing: special issue on field programmable analog arrays**. Dordrecht, The Netherlands: Kluwer Academic Publishers. v. 17, n. 1-2, p. 67-89, 1998.
- CALVANO, J. V.; ALVES, V. C.; LUBASZEWSKI, M. S. Fault Detection in Systems With 2<sup>nd</sup> Order Dynamics Using Transient Analysis. In: SYMPOSIUM ON INTEGRATED CIRCUITS AND SYSTEMS DESIGN, 1999, Natal, Brasil. **Proceedings...**[S. l.: S. n.], 1999, p.110 – 114.
- CALVANO, J. V.; ALVES, V. C.; LUBASZEWSKI, M. S. Fault detection methodology and BIST method for 2<sup>nd</sup> order Butterworth, Chebyshev and Bessel filter approximations. In: VLSI TEST SYMPOSIUM, 18., 2000, Montreal, Canada. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 2000, p. 319-324.
- CALVANO, J. V. *et al.* Filters Designed for Testability Wrapped on the Mixed-Signal Test Bus. In: VLSI TEST SYMPOSIUM, 20., 2002, Monterey, USA. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, may. 2002, p. 201-206.
- CALVANO, J. V. Transient Response Analysis Method Application. **Contato Pessoal**, Rio De Janeiro, Brasil, 2005.
- CHATTERJEE, A.; NAGI, N. Design for Testability and Built-In Self-Test of Mixed-Signal Circuits: a tutorial. In: INTERNATIONAL CONFERENCE ON VLSI DESIGN, 10., 1997, Hyderabad, India. **Proceedings...** [S. l.: S. n.], jan. 1997, p. 388 – 392.

CYPRESS Microsystems. CY8C2XXXX Family **Datasheet**, 2002. Disponível em [www.cypress.com](http://www.cypress.com). Acesso em: 8 mar. 2006.

DARYANANI, G. **Principles of Active Network Synthesis and Design**. New York, USA: John Wiley & Sons, 1976.

DOERNBERG, J.; LEE, H.; HODGES, D. Full-Speed Testing of A/D Converters. **IEEE Journal of Solid-State Circuits**, New York, USA: IEEE, v. 19, n. 6, p. 820-827, dec. 1984.

EETIMES. **EETIMES Electronic Magazine**. 2000. Disponível em <http://www.eetimes.com/semi/news/OEG20000731S0026>. Acesso em: 10 apr. 2005.

FLORES, M. G. **Teste Embarcado de Conversores Analógico-Digitais**, 2003. Dissertação (Mestrado em Engenharia Elétrica), Programa de Pós-graduação em Engenharia Elétrica, Universidade Federal do Rio Grande do Sul, 2003.

GAUDET, V.C.; GULAK, P.G. CMOS Implementation of a Current Conveyor-Based Field-Programmable Analog Array. In: ASILOMAR CONFERENCE ON SIGNALS, SYSTEMS & COMPUTERS, 31., 1997, Pacific Grove, USA. **Conference Record...** [S. l.: S. n.], v.2, nov. 1997, p.1156-1159.

HAN, D. *et al.* On-Chip Self-Calibration of RF Circuits Using Specification-Driven Built-In Self Test (S-BIST). In: IEEE INTERNATIONAL ON-LINE TESTING SYMPOSIUM, 11., 2005, Saint-Raphael, France. **Proceedings...** [S. l.: S. n.], 2005, p. 106 – 111.

HAYKIN, S.; VAN VEEN, B. **Sinais e Sistemas**. Rio de Janeiro, Brasil: Bookman do Brasil, 2001.

HUGHES, J. L. A. Multiple Fault Detection Using Single Fault Test Sets. **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**, New York, USA: IEEE, v. 7, n. 1, p.100 – 108, jan. 1988.

HUERTAS, G. *et al.* Testing Mixed-Signal Cores: a practical oscillation-based test in an analog macrocell. **IEEE Design & Test of Computers**, Los Alamitos, USA: IEEE Computer Society Press, v. 19, n. 6, nov. 2002 p. 64-72.

HUERTAS, G. *et al.* Practical Oscillation-Based Test in Analog Integrated Filters: experimental results. In: IEEE INTERNATIONAL WORKSHOP ON ELECTRONIC DESIGN, TEST AND APPLICATIONS, 1., 2002, Christchurch, New Zeland. **Proceedings...** [S. l.: S. n.], jan. 2002b, p.18 – 24.

HEREFORD, J.; PRUITT, C. Robust Sensor Systems Using Evolvable Hardware. In: NASA/DoD CONFERENCE ON EVOLVABLE HARDWARE, 2004, Seattle, USA. **Proceedings...** [S. l.: S. n.], 2004 p. 161.

IEEE Standards Association. IEEE Standard Test Access Port and Boundary-Scan Architecture. **IEEE Standard 1149.1**. New York, USA: IEEE, 1990. Disponível em: [http://standards.ieee.org/reading/ieee/std\\_public/description/testtech/](http://standards.ieee.org/reading/ieee/std_public/description/testtech/) Acesso em: 28 june, 2006.

IEEE Standards Association. IEEE Standard for a Mixed-Signal Test Bus. **IEEE Standard 1149.4**. New York, USA: IEEE, 1990. Disponível em: [http://standards.ieee.org/reading/ieee/std\\_public/description/testtech/](http://standards.ieee.org/reading/ieee/std_public/description/testtech/) Acesso em: 28 june, 2006.

KAC, U. *et al.* Extending IEEE Std. 1149.4 Analog Boundary Modules to Enhance Mixed-Signal Test. **IEEE Design & Test of Computers**, Los Alamitos, USA: IEEE Computer Society Press, v. 20, n. 2, p. 32-39, mar. 2003.

KONEMANN, B.; MUCHA, J.; ZWIEHOFF, G. Built-In Logic Block Observation Techniques. In: IEEE TEST CONFERENCE, 1979, Cherry Hill, USA. **Proceedings...** [S. l.: S. n.], 1979.

KUTUK, H.; KANG, S.M. A Switched Capacitor Approach to Field-Programmable Analog Array (FPAA) Design. **Analog Integrated Circuits and Signal Processing**: special issue on field programmable analog arrays. Dordrecht, The Netherlands: Kluwer Academic Publishers. v. 17, n. 1-2, p. 51-65, 1998.

LATTICE. **Programmable Analog Circuits: ispPAC handbook**. Hillsboro, USA: Lattice Semiconductor Corporation, 2000.

LEE, E.; GULAK, G. A CMOS Field-Programmable Analog Array. **IEEE Journal of Solid-State Circuits**, New York, USA: IEEE, v.26, n.12, p.1860-1867, dec. 1991.

LEE, E.; GULAK, G. Field Programmable Analogue Array Based on MOSFET Transconductors. **Electronics Letters**, London, UK: IEE, v.28, n.1, p.28-29, jan. 1992.

LEE, E.; GULAK, G. A Transconductor-Based Field-Programmable Analog Array. In: IEEE INTERNATIONAL SOLID-STATE CIRCUITS CONFERENCE, 42., 1995, San Francisco, USA. **Digest of Technical Papers...** [S. l.: S. n.], feb. 1995, p.198-199.

LEE, E.K.F.; HUI, W.L. A Novel Switched-Capacitor Based Field-Programmable Analog Array Architecture. **Analog Integrated Circuits and Signal Processing**: special issue on field programmable analog arrays. Dordrecht, The Netherlands: Kluwer Academic Publishers. v. 17, n. 1-2, p. 35-50, 1998.

LEHTONEN, T.; POSILA, J.; ISOAHO, J. On Fault Tolerance Techniques Towards Nanoscale Circuits and Systems. **Turku Centre for Computer Science Technical Report**, Turku, Finland, n. 708, aug. 2005.

LUBASZEWSKI, M.; MIR, S.; PULZ, L. ABILBO: analog built-in block observer. In: IEEE/ACM INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN, 1996, [S. l.]. **Proceedings...** [S. l.: S. n.], 1996, p. 600-603.

LUBASZEWSKI, M. *et al.* Design of Self-Checking Fully Differential Circuits and Boards. **IEEE Transactions on Very Large Scale Integration (VLSI) Systems**, New York, USA: IEEE, v.2, n. 2, p. 113, apr. 2000.

LUBASZEWSKI, M.; COTA, E.; KRUG, M. Teste e Projeto Visando o Teste de Circuitos e Sistemas Integrados. In: REIS, R. **Concepção de Circuitos Integrados**. Porto Alegre, Brasil: Sagra Luzzatto, 2002, p. 167-189.

MESSENGER, G. C. A Summary Review of Displacement Damage from High Energy Radiation in Silicon Semiconductors and Semiconductors Devices. **IEEE Transactions on Nuclear Science**, New York, USA: IEEE, v. 39, n. 3, p. 468-473, june 1992.

MILOR, L.; VISVANATHAN, V. Efficient Go/No Go Testing of Analog Circuits. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, 1987, [S. l.]. **Proceedings...** [S. l.: S. n.], 1987 p. 414-417.

MILOR, L.; VISVANATHAN, V. Detection of Catastrophic Faults in Analog Integrated Circuits. **IEEE Transactions on Computer-Aided Design**, New York, USA: IEEE, v. 8, n. 2, p. 114-130, feb. 1989.

MIR, S.; LUBASZEWSKI, M.; COURTOIS, B. Fault-Based ATPG for Linear Analog Circuits with Minimal Size Multifrequency Test Sets. **Journal of Electronic Testing: theory and applications**, special issue on mixed-signal testing, Dordrecht, The Netherlands: Kluwer Academic Publishers, v. 9, p. 43-57, aug. 1996.

MOORE, G. E. Cramming More Components Onto Integrated Circuits. **Electronics Magazine**, [S. l.: S. n.], v. 38, n. 8, apr.1965.

NEGREIROS, M.; CARRO, L.; SUSIN, A. A. A Statistical Sampler for a New On-line Analog Test Method. In: IEEE INTERNATIONAL ON-LINE TESTING WORKSHOP, 8., 2002, Isle of Bendor, France. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 2002, p. 79-83.

NEGREIROS, M.; CARRO, L.; SUSIN, A. A. A Low Cost On-Line Testing of RF Circuits In: IEEE INTERNATIONAL ON-LINE TESTING SYMPOSIUM, 10., 2004, Madeira Island, Portugal. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 2004, p. 73-78.

OGATA, K. **Engenharia de Controle Moderno**. Rio de Janeiro: Prentice/Hall do Brasil, 1982.

OSSEIRAN, A. IEEE 1149.4 Analog and Mixed-Signal Test Bus Standard. IEEE EUROPEAN TEST WORKSHOP, 2000, Cascais, Portugal. 1999. Disponível em [http://www.lirmm.fr/~w3mic/ETW/ETW00/Program/Session6A/etw6A\\_1.pdf](http://www.lirmm.fr/~w3mic/ETW/ETW00/Program/Session6A/etw6A_1.pdf). Acesso em: 3 july, 2006.

PERALÍAS, E. J.; RUEDA, A.; HUERTAS, J. L. Structural Testing of Pipelined Analog to Digital Converters. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, 2001, [S. l.]. **Proceedings...** [S. l.: S. n.], may. 2001, p. 436 – 439.

PEREIRA, G. *et al.* Testing the Interconnect Networks and I/O Resources of Field Programmable Analog Arrays. In: IEEE VLSI TEST SYMPOSIUM, 23., 2005, Palm Springs, USA. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 2005, p. 389 – 394.

PERETTI *et al.* Using Oscillation Based Test for Testing Digital Spectrometers. In: IEEE LATIN-AMERICAN TEST WORKSHOP, 6., 2005, Salvador, Brasil. **Digest of Papers...**[S. l.: S.n.], mar. 2005, p. 271-276.

PREMONT, C. *et al.* A Current Conveyor Based Field Programmable Analog Array. **Analog Integrated Circuits and Signal Processing**: special issue on field programmable analog arrays. Dordrecht, The Netherlands: Kluwer Academic Publishers. v. 17, n. 1-2, p.105-124, sept. 1998.

RENOVELL, M. Digital and Analog System Testing: fundamentals and new challenges In: The 16th INTERNATIONAL CONFERENCE ON MICROELECTRONICS, 2004, Tunis, Tunisia. **Proceedings...** [S. l.: S. n.], 2004, p. 8 – 10.

SEDRA, A.S.; ROBERTS, G.W.; GOHH, F. The Current Conveyor: history, progress and new results. **IEEE Proceedings on Circuits, Devices and Systems**, New York, USA: IEEE, v.137, n.2, p.78-87, apr. 1990.

SEDRA, A.; SMITH, K. **Microelectronic Circuits**. Orlando, USA: Saunders College Publishing, 1991.

SOMA, M. Challenges in Analog and Mixed-Signal Fault Models. **IEEE Circuits and Devices Magazine**, New York, USA: IEEE, v. 12, p. 16 – 19, jan. 1996.

SOUDERS, T. M.; STENBAKKEN, G. N. A Comprehensive Approach for Modeling and Testing Analog and Mixed-Signal Devices. In: INTERNATIONAL TEST CONFERENCE, 1990, Washington DC, USA. **Proceedings...** Washington DC, USA: International Test Conference Press, 1990, p. 169 – 176.

STAPPER, C. H.; ARMSTRONG, F. M.; SAJI, K. Integrated Circuit Yield Statistics, In: **Proceedings of the IEEE**, New York, USA: IEEE, v. 71, abr. 1983, p. 453-470.

SUNTER, S.; NAGI, N. Test Metrics for Analog Parametric Faults. In: IEEE VLSI TEST SYMPOSIUM, 17., 1999, San Diego, USA. **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 1999, p. 226 – 234.

TAUR, Y. *et al.* CMOS Scaling into the Nanometer Regime. **Proceedings of the IEEE**. New York, USA: IEEE, v. 85, n. 4, apr. 1997, p. 486 – 504.

VÁZQUZ, D.; RUEDA, A.; HUERTAS, J. L. A practical implementation of fault-tolerant switched-capacitor circuits. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, 1991, [S. l.]. **Proceedings...** [S. l.: S. n.], jun. 1991, v.3, p.1565 – 1568.

VÁZQUEZ, *et al.* On-chip Evaluation of Oscillation-Based Test Output Signals for Switched-Capacitor Circuits. **Analog Integrated Circuits and Signal Processing**, Dordrecht, The Netherlands: Kluwer Academic Publishers, v. 33, n. 2, p. 201-211, 2002.

VÁZQUEZ, *et al.* A Simple and Secure Start-Up Circuitry for Oscillation-Based Test Application. **Analog Integrated Circuits and Signal Processing**, Dordrecht, The Netherlands: Kluwer Academic Publishers, v. 32, n.2, p. 187-190, 2002b.

VELASCO-MEDINA, J.; NICOLAIDIS, M.; LUBASZEWSKI, M. An Approach to the On-Line Testing of Operational Amplifiers. In: ASIAN TEST SYMPOSIUM, 7., Singapore, 1998, **Proceedings...** Los Alamitos, USA: IEEE Computer Society Press, 1998, p. 290 – 295.

ZETEX. TRAC020LH – Totally Re-Configurable Analog Circuit – TRAC. **DataSheet**, 1999. Disponível em: [www.zetex.com.uk](http://www.zetex.com.uk). Acesso em: 10 mar. 2006.

ZHANG, C.; BRATT, A.; MACBETH, I. A New Field Programmable Mixed Signal Array and its Applications. In: CANADIAN WORKSHOP ON FIELD PROGRAMMABLE DEVICES, 1996, Toronto, Canada. **Proceedings...** [S. l.: S. n.], mai. 1996.

ZNAMIROWSKI, L; PAULUSINSKI, O. A.; VRUDHULA, S. B. K. Programmable Analog/Digital Arrays in Control and Simulation. **Analog Integrated Circuits and Signal Processing**, Dordrecht, The Netherlands: Kluwer Academic Publishers, v. 39, n.1, p. 55–73, 2004.

**APÊNDICE A: CÓDIGO EM *MATLAB* PARA CÁLCULO DA SENSIBILIDADE DOS  
PARÂMETROS DA RESPOSTA AO *STEP* EM RELAÇÃO AOS PARÂMETROS DAS FUNÇÕES DE  
TRANSFERÊNCIA DE 1<sup>A</sup> E 2<sup>A</sup> ORDEM**

%Análise de sensibilidade sobre variação da resposta ao step de filtros de primeira e segunda

%ordem relativo aos seus parâmetros das funções de transferência.

%definição de variáveis simbólicas:

fo1=sym('fo1');

fo2=sym('fo2');

Q=sym('Q');

%Definição das relações

%primeira ordem

%Tempo de acomodação

Ts1=4/fo1;

%definições

%x e o fator de amortecimento

$x = 1/(2*Q)$ ;

% wd e a frequência amortecida

$wd=fo2*((1-x^2)^{.5})$ ;

%segunda ordem

%Tempo de acomodação

$Ts2=4/(fo2*x)$ ;

% tempo de pico

$Tp=pi/wd$ ;

% Overshoot

$Os = \exp(-pi*(x/((1-x^2)^{0.5})))$ ;

%Cálculo das derivadas

dTs1\_dfo1=diff(Ts1,'fo1');

dTs2\_dfo2=diff(Ts2,'fo2');

dTs2\_dQ=diff(Ts2,'Q');

dTp\_dfo2=diff(Tp,'fo2');

dTp\_dQ=diff(Tp,'Q');

dOs\_dQ=diff(Os,'Q');

%calculo das sensibilidades

$$STs1\_fo1 = (fo1/Ts1)*dT_s1\_dfo1$$

$$STs2\_fo2 = (fo2/Ts2)*dT_s2\_dfo2$$

$$STs2\_Q = (Q/Ts2)*dT_s2\_dQ$$

$$STp\_fo2 = (fo2/Tp)*dT_p\_dfo2$$

$$STp\_Q = (Q/Tp)*dT_p\_dQ;$$

$$SOs\_Q = (Q/Os)*dOs\_dQ;$$

$$STp\_Q = \text{simplify}(STp\_Q)$$

$$SOs\_Q = \text{simplify}(SOs\_Q)$$

**APÊNDICE B: CÓDIGO EM *MATLAB* PARA CÁLCULO DA SENSIBILIDADE DOS  
PARÂMETROS DAS FUNÇÕES DE TRANSFERÊNCIA DE 1<sup>A</sup> E 2<sup>A</sup> ORDEM A VARIAÇÕES NOS  
CAPACITORES PROGRAMÁVEIS DOS CABS**

```

%Análise de sensibilidade dos parâmetros das funções de transferência de primeira e
segunda
%ordem em relação aos componentes programáveis dos blocos. Considerando CAB
%full-diferencial

%definição de variáveis simbólicas:
C1p=sym('C1p');
C2p=sym('C2p');
C3p=sym('C3p');
C4p=sym('C4p');
CAp=sym('CAp');
CBp=sym('CBp');
C1m=sym('C1m');
C2m=sym('C2m');
C3m=sym('C3m');
C4m=sym('C4m');
CAm=sym('CAm');
CBm=sym('CBm');

fc=sym('fc');

%Definição das funções

%primeira ordem
%ganho
G1=((C1p/C3p)+(C1m/C3m))/2;
%freq. de corte
fo1=(fc/pi)*(C3p*C3m/(4*(C2p*C2m)+2*(C2p*C3m)+2*(C3p*C2m)+(C3p*C3m)))^0
.5;

%segunda ordem
%ganho
G2=((C1p/C2p)+(C1m/C2m))/2;

fo2=(fc/(2*pi))*((((C2p*C3p)/(CAp*CBp))^0.5)*(((C2m*C3m)/(CAm*CBm))^0.5))^0.5;
Q=(((CBp/C4p)*sqrt(C2p*C3p/CAp*CBp))+((CBm/C4m)*sqrt(C2m*C3m/CAm*CBm
)))/2;

%Cálculo das derivadas

```

```

dG1_dC1p=diff(G1,'C1p');
dG1_dC3p=diff(G1,'C3p');
dG1_dC1m=diff(G1,'C1m');
dG1_dC3m=diff(G1,'C3m');

```

```

dfo1_dC2p=diff(fo1,'C2p');
dfo1_dC3p=diff(fo1,'C3p');
dfo1_dfc=diff(fo1,'fc');
dfo1_dC2m=diff(fo1,'C2m');
dfo1_dC3m=diff(fo1,'C3m');

```

```

dG2_dC1p=diff(G2,'C1p');
dG2_dC2p=diff(G2,'C2p');
dG2_dC1m=diff(G2,'C1m');
dG2_dC2m=diff(G2,'C2m');

```

```

dfo2_dC2p=diff(fo2,'C2p');
dfo2_dC3p=diff(fo2,'C3p');
dfo2_dCAp=diff(fo2,'CAp');
dfo2_dCBp=diff(fo2,'CBp');
dfo2_dfc=diff(fo2,'fc');
dfo2_dC2m=diff(fo2,'C2m');
dfo2_dC3m=diff(fo2,'C3m');
dfo2_dCAm=diff(fo2,'CAm');
dfo2_dCBm=diff(fo2,'CBm');

```

```

dQ_dC2p=diff(Q,'C2p');
dQ_dC3p=diff(Q,'C3p');
dQ_dC4p=diff(Q,'C4p');
dQ_dCAp=diff(Q,'CAp');
dQ_dCBp=diff(Q,'CBp');
dQ_dC2m=diff(Q,'C2m');
dQ_dC3m=diff(Q,'C3m');
dQ_dC4m=diff(Q,'C4m');
dQ_dCAm=diff(Q,'CAm');
dQ_dCBm=diff(Q,'CBm');

```

%calculo das sensibilidades

```

SG1_C1p= (C1p/G1)*dG1_dC1p;
SG1_C3p= (C3p/G1)*dG1_dC3p;
SG1_C1m= (C1m/G1)*dG1_dC1m;
SG1_C3m= (C3m/G1)*dG1_dC3m;

```

```

Sfo1_C2p= (C2p/fo1)*dfo1_dC2p;
Sfo1_C3p= (C3p/fo1)*dfo1_dC3p;
Sfo1_fc= (fc/fo1)*dfo1_dfc;
Sfo1_C2m= (C2m/fo1)*dfo1_dC2m;

```

$$\text{Sfo1\_C3m} = (\text{C3m}/\text{fo1}) * \text{dfo1\_dC3m};$$

$$\begin{aligned} \text{SG1\_C1p} &= \text{simplify}(\text{SG1\_C1p}) \\ \text{SG1\_C3p} &= \text{simplify}(\text{SG1\_C3p}) \\ \text{SG1\_C1m} &= \text{simplify}(\text{SG1\_C1m}) \\ \text{SG1\_C3m} &= \text{simplify}(\text{SG1\_C3m}) \\ \text{Sfo1\_C2p} &= \text{simplify}(\text{Sfo1\_C2p}) \\ \text{Sfo1\_C3p} &= \text{simplify}(\text{Sfo1\_C3p}) \\ \text{Sfo1\_C2m} &= \text{simplify}(\text{Sfo1\_C2m}) \\ \text{Sfo1\_C3m} &= \text{simplify}(\text{Sfo1\_C3m}) \end{aligned}$$

$$\begin{aligned} \text{SG2\_C1p} &= (\text{C1p}/\text{G2}) * \text{dG2\_dC1p}; \\ \text{SG2\_C2p} &= (\text{C2p}/\text{G2}) * \text{dG2\_dC2p}; \\ \text{SG2\_C1m} &= (\text{C1m}/\text{G2}) * \text{dG2\_dC1m}; \\ \text{SG2\_C2m} &= (\text{C2m}/\text{G2}) * \text{dG2\_dC2m}; \end{aligned}$$

$$\begin{aligned} \text{Sfo2\_C2m} &= (\text{C2m}/\text{fo2}) * \text{dfo2\_dC2m} \\ \text{Sfo2\_C3m} &= (\text{C3m}/\text{fo2}) * \text{dfo2\_dC3m} \\ \text{Sfo2\_CAm} &= (\text{CAm}/\text{fo2}) * \text{dfo2\_dCAm} \\ \text{Sfo2\_CBm} &= (\text{CBm}/\text{fo2}) * \text{dfo2\_dCBm} \\ \text{Sfo2\_fc} &= (\text{fc}/\text{fo2}) * \text{dfo2\_dfc} \\ \text{Sfo2\_C2p} &= (\text{C2p}/\text{fo2}) * \text{dfo2\_dC2p} \\ \text{Sfo2\_C3p} &= (\text{C3p}/\text{fo2}) * \text{dfo2\_dC3p} \\ \text{Sfo2\_CAp} &= (\text{CAp}/\text{fo2}) * \text{dfo2\_dCAp} \\ \text{Sfo2\_CBp} &= (\text{CBp}/\text{fo2}) * \text{dfo2\_dCBp} \end{aligned}$$

$$\begin{aligned} \text{SG2\_C1p} &= \text{simplify}(\text{SG2\_C1p}) \\ \text{SG2\_C2p} &= \text{simplify}(\text{SG2\_C2p}) \\ \text{SG2\_C1m} &= \text{simplify}(\text{SG2\_C1m}) \\ \text{SG2\_C2m} &= \text{simplify}(\text{SG2\_C2m}) \end{aligned}$$

$$\begin{aligned} \text{SQ\_C2p} &= (\text{C2p}/\text{Q}) * \text{dQ\_dC2p}; \\ \text{SQ\_C3p} &= (\text{C3p}/\text{Q}) * \text{dQ\_dC3p}; \\ \text{SQ\_C4p} &= (\text{C4p}/\text{Q}) * \text{dQ\_dC4p}; \\ \text{SQ\_CAp} &= (\text{CAp}/\text{Q}) * \text{dQ\_dCAp}; \\ \text{SQ\_CBp} &= (\text{CBp}/\text{Q}) * \text{dQ\_dCBp}; \\ \text{SQ\_C2m} &= (\text{C2m}/\text{Q}) * \text{dQ\_dC2m}; \\ \text{SQ\_C3m} &= (\text{C3m}/\text{Q}) * \text{dQ\_dC3m}; \\ \text{SQ\_C4m} &= (\text{C4m}/\text{Q}) * \text{dQ\_dC4m}; \\ \text{SQ\_CAm} &= (\text{CAm}/\text{Q}) * \text{dQ\_dCAm}; \\ \text{SQ\_CBm} &= (\text{CBm}/\text{Q}) * \text{dQ\_dCBm}; \end{aligned}$$

$$\begin{aligned} \text{SQ\_C2p} &= \text{simplify}(\text{SQ\_C2p}) \\ \text{SQ\_C3p} &= \text{simplify}(\text{SQ\_C3p}) \\ \text{SQ\_C4p} &= \text{simplify}(\text{SQ\_C4p}) \\ \text{SQ\_CAp} &= \text{simplify}(\text{SQ\_CAp}) \\ \text{SQ\_CBp} &= \text{simplify}(\text{SQ\_CBp}) \\ \text{SQ\_C2m} &= \text{simplify}(\text{SQ\_C2m}) \\ \text{SQ\_C3m} &= \text{simplify}(\text{SQ\_C3m}) \end{aligned}$$

```
SQ_C4m= simplify(SQ_C4m)
SQ_CAm= simplify(SQ_CAm)
SQ_CBm= simplify(SQ_CBm)
```

## ANEXO A: CARACTERÍSTICAS ELÉTRICAS DO FPAA ISPPAC10 DA LATTICE

Specifications *ispPAC10*

$T_A = 25^\circ\text{C}$ ;  $V_S = 5.0\text{V}$ ; Signal path =  $V_{IN}$  to  $V_{OUT}$  of one PACblock (second input unused);  $1\text{V} \leq V_{OUT} \leq 4\text{V}$ ; Gain = 1; Output load = 200pF, 1M $\Omega$ . Feedback enabled; Feedback capacitor = minimum; Auto-Cal initiated immediately prior. (Unless otherwise specified).

## DC Electrical Characteristics

SYMBOL	PARAMETER	CONDITION	MIN.	TYP.	MAX.	UNITS
<b>Analog Input</b>						
$V_{IN\pm}$ (1)	Input Voltage Range	Applied Either to $V_{IN+}$ or $V_{IN-}$	1		4	V
$V_{IN-DIFF}$	Differential Input Voltage Swing (2)	2  $V_{IN+} - V_{IN-}$	6			V <sub>p-p</sub>
$V_{OS}$ (2)	Differential Offset Voltage (Input Referred)	G = 10 G = 1		20 0.2	100 1.0	$\mu\text{V}$ mV
$\Delta V_{OS}/\Delta T$	Differential Offset Voltage Drift	-40 to +85°C		50		$\mu\text{V}/^\circ\text{C}$
$R_{IN}$	Input Resistance			10 <sup>9</sup>		$\Omega$
$C_{IN}$	Input Capacitance			2		pF
$I_B$	Input Bias Current	at DC		3		pA
$e_{n}$	Input Noise Voltage Density	At 10kHz, Referred to Input, G = 10		38		nV/ $\sqrt{\text{Hz}}$
<b>Analog Output</b>						
$V_{OUT\pm}$	Output Voltage Range	Present at Either $V_{OUT+}$ or $V_{OUT-}$	0.1		4.9	V
$V_{OUT-DIFF}$	Differential Output Voltage Swing (2)	2  $V_{OUT+} - V_{OUT-}$	9.6			V <sub>p-p</sub>
$I_{OUT\pm}$	Output Current	Source/Sink	10			mA
$V_{CM}$	Common Mode Output Voltage	$(V_{OUT+} + V_{OUT-})/2$ ; $V_{IN+} = V_{IN-}$	2.495	2.500	2.505	V
<b>Static Performance</b>						
G	Programmable Gain Range	Each Individual PACblock	0		20	dB
	Gain Error	$R_L = 300\Omega$ Differential			4.0	%
	Gain Matching	Between Two Inputs of Same PACblock			3.0	%
$\Delta G/\Delta T$	Gain Drift	-40 to +85°C		20		ppm/°C
PSR	Power Supply Rejection	Differential at 1kHz Single-ended at 1kHz		80 77		dB dB
<b>Common Mode Reference Output (VREF<sub>OUT</sub>)</b>						
$V_{REF-OUT}$	Reference Output Voltage Range	Nominally 2.500V	-0.2		0.2	%
$CMV_{IN}$ (4)	Common Mode Voltage Input	Optional External Common-Mode Voltage	1.25		3.25	V
	Reference Output Voltage Drift	-40 to +85°C		50		ppm/°C
$I_{REF-OUT}$	Reference Output Current	( $V_{REF-OUT} = \pm 1\%$ ) Source ( $V_{REF-OUT} = \pm 1\%$ ) Sink		50 350		$\mu\text{A}$ $\mu\text{A}$
	Reference Output Noise Voltage	10MHz Bandwidth; 1 $\mu\text{F}$ Bypass Capacitor		40		$\mu\text{V}_{RMS}$
	Reference Power Supply Rejection	1kHz		80		dB
<b>Programming</b>						
	Erase/Reprogram Cycles		10K			cycles
<b>Digital I/O</b>						
$V_{IL}$	Input Low Voltage		0		0.8	V
$V_{IH}$	Input High Voltage		2.0		$V_S$	V
$I_L, I_H$	Input Leakage Current	0V $\leq$ TCK Input $\leq V_S$ 0V $\leq$ CAL, TDI, TMS, TRST Inputs $\leq V_S$			$\pm 10$ +40/-70	$\mu\text{A}$ $\mu\text{A}$
$V_{OL}$	Output Low Voltage (TDO)	$I_{OL} = 4.0\text{mA}$			0.5	V
$V_{OH}$	Output High Voltage (TDO)	$I_{OH} = -1.0\text{mA}$	2.4			V
<b>Power Supplies</b>						
$V_S$	Operating Supply Voltage		4.75	5.0	5.25	V
$I_S$	Supply Current	$V_S = 5.0\text{V}$			23	mA
$P_D$	Power Dissipation	$V_S = 5.0\text{V}$			115	mW
<b>Temperature Range</b>						
	Operation		-40		+85	°C
	Storage		-65		+150	°C

## ANEXO B: CARACTERÍSTICAS ELÉTRICAS DO FPAA AN10E40 DA ANADIGM

### Absolute Maximum Ratings

	Min.	Typ.	Max.	Notes
Supply Voltages (A,B,D,SVDD)	-0.5 V		6.5 V	1
Analog Input Voltage	-0.5 V		AVDD+0.5V	
Digital Input Voltage	-0.5 V		DVDD+0.5V	
Storage Temperature	-85 C		150 C	

1 - Operation with V<sub>dd</sub> > 5.5 V may reduce device operating lifetime.

### Recommended Operating Conditions

	Min.	Typ.	Max.	Notes
Supply Voltages (A,B,D,SVDD)	4.5 V	5.0 V	5.5 V	
Analog Input Voltage	0.5 V		AVDD-0.5V	
Standard Analog Load (small signal)		1k $\Omega$    100pF		1 k $\Omega$ in parallel with 100 pF
Standard Analog Load (large signal)		10k $\Omega$    100pF		10 k $\Omega$ in parallel with 100 pF
Standard Digital Load		50 pF		50 pF to DVSS
Ambient Operating Temperature	-40 C		+85 C	

### Digital IO

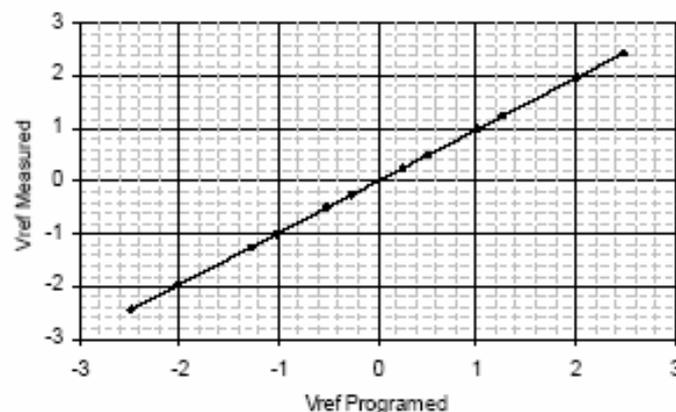
	Min.	Typ.	Max.	Notes
Output Voltage High (V <sub>oh</sub> )	0.8 V <sub>dd</sub>			
Input High Voltage (V <sub>ih</sub> )	0.7 V <sub>dd</sub>			
Input Low Voltage (V <sub>il</sub> )			0.3 V <sub>dd</sub>	
Output Voltage Low (V <sub>ol</sub> )			0.2 V <sub>dd</sub>	
Tri-State Leakage Current (I <sub>oh</sub> or I <sub>ol</sub> )			negligible	

### Voltage Mid Rail

The array supplies its own internal analog ground reference known as VMR. VMR is 2.5 V above AVSS. Noise on VMR degrades system performance so great care has been taken to provide the AN10E40 with an extremely quiet analog reference generator.

	Min.	Typ.	Max.	Notes
VMR		2.5 V		

### Vref



## ANEXO C: CARACTERÍSTICAS ELÉTRICAS DO FPAA AN221E04 DA ANADIGM

### AN221E04 Datasheet – Dynamically Reconfigurable FPAA With Enhanced I/O

#### ELECTRICAL CHARACTERISTICS

##### Absolute Maximum Ratings

Parameter	Symbol	Min	Typ	Max	Unit	Comment
DC Power Supplies	AVDD(2) BVDD DVDD	-0.5	-	5.5 V	V	AVSS, BVSS, DVSS and SVSS all held to 0.0 V <sup>a</sup>
xVDD to xVDD Offset		-0.5		0.5	V	Ideally all supplies should be at the same voltage
Package Power Dissipation	P <sub>max</sub> 25°C P <sub>max</sub> 85°C	-	-	1.8 0.73	W	Still air, No heatsink, 4 layer board, 44 pins. $\theta_{ja} = 55^{\circ}\text{C/W}$
Analog and Digital Input Voltage	V <sub>inmax</sub>	V <sub>ss</sub> -0.5	-	V <sub>dd</sub> +0.5	V	
Ambient Operating Temperature	Top	-40	-	85	°C	
Storage Temperature	T <sub>stg</sub>	-65		150	°C	

<sup>a</sup> Absolute Maximum DC Power Supply Rating - The failure mode is non-catastrophic for V<sub>dd</sub> of up to 7 volts, but will cause reduced operating life time. The additional stress caused by higher local electric fields within the CMOS circuitry may induce metal migration, oxide leakage and other time/quality related issues.

##### Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Unit	Comment
DC Power Supplies	AVDD(2) BVDD DVDD	4.75	5.00	5.25	V	AVSS, BVSS, DVSS and SVSS all held to 0 V
Analog Input Voltage	V <sub>ina</sub>	V <sub>MR</sub> -1.9	-	V <sub>MR</sub> +1.9	V	V <sub>MR</sub> is 2.0 volts above AVSS
Digital Input Voltage	V <sub>ind</sub>	0	-	DVDD	V	
Junction Temp	T <sub>J</sub>	-40	-	125	°C	Assume a package $\theta_{ja} = 55^{\circ}\text{C/W}$ <sup>b</sup>

<sup>b</sup> In order to calculate the junction temperature you must first empirically determine the current draw (total I<sub>dd</sub>) for the design. Once the current consumption established then the following formula can be used;  $T_J = T_a + I_{dd} \times V_{dd} \times 55^{\circ}\text{C/W}$ , where T<sub>a</sub> is the ambient temperature. The worst case  $\theta_{ja}$  of 55 °C/W assumes no air flow and no additional heatsink of any type.

##### General Digital I/O Characteristics (V<sub>dd</sub> = 5v +/- 10%, -40 to 85 deg.C)

Parameter	Symbol	Min	Typ	Max	Unit	Comment
Input Voltage Low	V <sub>ih</sub>	0	-	30	-	% of DVDD
Input Voltage High	V <sub>il</sub>	70	-	100	-	% of DVDD
Output Voltage Low	V <sub>ol</sub>	0	-	20	-	% of DVDD
Output Voltage High	V <sub>oh</sub>	80	-	100	-	% of DVDD
Input Leakage Current	I <sub>il</sub>	-	-	±1.0	µA	All pins except DCLK
Input Leakage Current	I <sub>il</sub>	-	±12.0	-	µA	DCLK if a crystal is connected and the on-chip oscillator is used
Max. Capacitive Load	C <sub>max</sub>	-	-	10	pF	The maximum load for a digital output is 10 pF // 10 Kohm
Min. Resistive Load	R <sub>min</sub>	10	-	-	Kohm	The maximum load for a digital output is 10 pF // 10 Kohm
DCLK Frequency	F <sub>max</sub>	-	-	40	MHz	For MODE = 1, Max DCLK is 16 MHz
ACLK Frequency	F <sub>max</sub>	-	-	40	MHz	Divide down to <8 MHz prior to use as a CAB clock
Clock Duty Cycle	-	45	-	55	%	All clocks

## ANEXO D: COMPARATIVO DAS CARACTERÍSTICAS GERAIS DOS FPAAS DA FAMÍLIA

### VORTEX DA ANADIGM

	ANADIGMVORTEX				
	AN221E02	AN120E04	AN121E04	AN220E04	AN221E04
Supply Voltage	5V	5V	5V	5V	5V
Input/Output Signals	Single-Ended OR Differential				
Core Architecture	Differential	Differential	Differential	Differential	Differential
Input DC Offset	<100µV	<100µV	<100µV	<100µV	<100µV
SNR (Target Only)	80dB Broadband 100dB Narrowband				
Signal Bandwidth (Application Dependent)	DC - 2MHz				
Cross Talk Rejection	>70 db				
Array Size	2x1 CAB Array	2x2 CAB Array	2x2 CAB Array	2x2 CAB Array	2x2 CAB Array
CAB Resources	2 Fully Differential Op Amps				
	8x Dual Capacitor Bank				
	8-bit Successive Approximation Register (SAR)				
	1 Hi-speed Comparator				
Additional Functions	LUT	LUT	LUT	LUT	LUT
	Programmable Counter				
	8-bit SAR Based A/D Converter (for use off-chip)		8-bit SAR Based A/D Converter (for use off-chip)		8-bit SAR Based A/D Converter (for use off-chip)
Data Load Time	1.8-120 µsec				
Inputs/Outputs	1 Input/Output	3 Dedicated Inputs	3 Input/Output	3 Dedicated Inputs	3 Input/Output
	1 Input/Output with a 4:1 Multiplexer	1 Input with a 4:1 Input Multiplexer	1 Input/Output with a 4:1 Multiplexer	1 Input with a 4:1 Input Multiplexer	1 Input/Output with a 4:1 Multiplexer
	2 Dedicated Outputs				
Reconfiguration Mode	Static and Dynamic	Static	Static	Static and Dynamic	Static and Dynamic
Pin Count/Package	44-pin QFP				
Package Dimensions (mm)	10x10x2	10x10x2	10x10x2	10x10x2	10x10x2