

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

**RAUL VIEIRA AMARAL**

**METODOLOGIA DE ANÁLISE DA VARIABILIDADE EM  
FPGA**

Porto Alegre

2010

**RAUL VIEIRA AMARAL**

**METODOLOGIA DE ANÁLISE DA VARIABILIDADE EM  
FPGA**

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: Tecnologia de Informação e Comunicações.

**ORIENTADOR:** Prof. Dr. Gilson Inácio Wirth

Porto Alegre

2010

**RAUL VIEIRA AMARAL**

# **METODOLOGIA DE ANÁLISE DA VARIABILIDADE EM FPGA**

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. Gilson Inácio Wirth, UFRGS

Doutor pela Universitaet Dortmund – Dortmund, Alemanha

Banca Examinadora:

Profª. Dra. Fernanda Gusmão de Lima Kastensmidt, UFRGS

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

Prof. Dr. Altamiro Amadeu Susin, UFRGS

Doutor pelo Institut National Polytechnique de Grenoble – Grenoble, França

Prof. Dr. Marcelo Götz - UFRGS

Doutor pela Universität Paderborn – Paderborn, Alemanha

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

Prof. Dr. Alexandre Sanfelice Bazanella

Porto Alegre, Junho de 2010.

## **DEDICATÓRIA**

Dedico este trabalho a minha esposa Loreci e ao meu filho Rodrigo, em especial pela dedicação e apoio em todos os momentos.

## **AGRADECIMENTOS**

Ao Programa de Pós-Graduação em Engenharia Elétrica, PPGEE, pela oportunidade de realização de trabalhos em minha área de pesquisa. Em especial, ao Prof. Dr. Roberto Petry Homrich por sua dedicação por esse MINTER e ao meu orientador Prof. Dr. Gilson Inácio Wirth principalmente, pelo seu caráter do qual destaco a firmeza e coerência de atitudes.

Ao pessoal do LaPSI, em especial, ao Prof. Dr. Altamiro Amadeu Susin, a Dra. Leticia Vieira Guimarães, o Dr. André Borin Soares, Dr. Marcelo Negreiros e Eng. M.Sc. Alexandro Cristovão Bonatto.

Aos colegas do IFSUL e do Curso Técnico em Telecomunicações pela oportunidade de realização desse curso e pelo auxílio nas tarefas desenvolvidas. Em especial ao Prof. M.Sc. Sergio Luiz Schubert Severo por disponibilizar, com responsabilidade, todos os equipamentos que solicitei e aos professores M.Sc. Claudio Enrique Fernández Rodriguez e M.Sc. Paulo Renato Avendano Motta.

Aos colegas do MINTER pelo auxílio nas tarefas desenvolvidas.

À CAPES pela provisão da bolsa de mestrado.

## RESUMO

Este trabalho visa propor uma metodologia de análise da variabilidade do tempo de atraso de propagação no FPGA. Para alcançar esse objetivo são utilizados três circuitos diferentes: o circuito 1 mede a diferença de atrasos de dois circuitos, o circuito 2 identifica o atraso menor de dois circuitos e, por fim, o terceiro circuito que consiste do oscilador em anel. Cada circuito foi avaliado individualmente numa estrutura BIST, implementada nos FPGA XC3S200-FT256 e EP2C35F672C6. Os métodos utilizados para análise dos dados foram a média móvel, o plano de mínimos quadrados e o teste t-student. A metodologia permitiu mostrar a variabilidade within-die e suas componentes sistêmica e randômica.

**Palavras-chaves:** Engenharia Elétrica. FPGA. Variabilidade. Tempo de Atraso de Propagação. *Delay. Within-die.*

## **ABSTRACT**

This work aims to propose a methodology of analysis of variability of propagation-delay time in FPGA. To achieve this goal three different circuits are implemented: the circuit 1 measures the delay difference of two logic paths, the circuit 2 identifies smallest delay of two logic paths, and finally the third circuit consists of a ring oscillator. Each circuit has been assessed individually in a BIST structure, implemented in FPGAs XC3S200-FT256 and EP2C35F672C6. The methods used for data analysis were the moving average, least-squares plane and the t-student test. The methodology has allowed to evaluate the within-die variability and its systemic and random components.

**Keywords: Electrical Engineering. FPGA. Variability. Propagation-delay Time. Delay. Within-die.**

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>16</b>
<b>2 LITERATURA EM CARACTERIZAÇÃO DE FPGA.....</b>	<b>21</b>
<b>3 FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>23</b>
<b>3.1 Variabilidade.....</b>	<b>23</b>
3.1.1 Classificação da variabilidade de processo .....	24
3.1.2 Variabilidade decorrente do roteamento .....	26
<b>3.2 Circuitos Utilizados .....</b>	<b>26</b>
3.2.1 Circuito 1.....	27
3.2.2 Circuito 2 .....	28
3.2.3 Circuito 3 – Oscilador em anel .....	32
<b>4 METODOLOGIA.....</b>	<b>34</b>
4.1 Introdução .....	34
4.2 Procedimento Experimental .....	36
4.3 Metodologia da Análise .....	39
<b>5 RESULTADOS E ANÁLISES .....</b>	<b>41</b>
5.1 Circuito 1 Implementado no FPGA 2XC3S200 .....	41
5.2 Circuito 1 Implementado no FPGA EP2C35F672.....	52
5.3 Circuito 2 Implementado no FPGA 2XC3S200 .....	62
5.4 Circuito 2 Implementado no FPGA EP2C35F672.....	73
5.5 Circuito Oscilador em Anel Implementado no FPGA 2XC3S200 .....	81
5.6 Circuito Oscilador em Anel Implementado no FPGA EP2C35F672.....	90
5.7 Análise e Discussão dos Resultados.....	98
5.8 Trabalhos Futuros .....	102
<b>6 CONCLUSÕES.....</b>	<b>103</b>
<b>REFERÊNCIAS .....</b>	<b>105</b>
<b>APÊNDICES .....</b>	<b>109</b>
<b>APÊNDICE A: CÓDIGOS FONTE DOS CIRCUITO IMPLEMENTADOS .....</b>	<b>109</b>
<b>A.1 Código Fonte do Circuito 1 para o Spartan 3 .....</b>	<b>109</b>
A.1.1 Arquivo main.vhd.....	109
A.1.2 Arquivo main.ucf.....	110
A.1.3 Arquivo package_main.vhd.....	110
A.1.4 Arquivo latch_nor.vhd .....	111
A.1.5 Arquivo circuito_teste.vhd.....	112
A.1.6 Arquivo circuito_comparador.vhd .....	112
A.1.7 Arquivo contador_assincrono_mod_32.vhd .....	113
A.1.8 Arquivo jk_borda_neg.vhd.....	114
A.1.9 Arquivo porta_inversora.vhd.....	114
A.1.10 Arquivo porta_nand.vhd .....	115
A.1.11 Arquivo porta_ne_3in.vhd.....	115
A.1.12 Arquivo porta_ou.vhd.....	115
<b>A.2 Código Fonte do Circuito 1 para o Cyclone II .....</b>	<b>116</b>

A.2.1	Arquivo main.vhd.....	116
A.2.2	Arquivo main.ucf.....	117
A.2.3	Arquivo package_main.vhd.....	117
A.2.4	Arquivo circuito_teste.vhd.....	118
A.2.5	Arquivo contador_assincrono_mod_64.vhd .....	119
<b>A.3</b>	<b>Código Fonte do Circuito 1 Modificado para o Spartan 3.....</b>	<b>120</b>
A.3.1	Arquivo main.vhd.....	120
A.3.2	Arquivo main.ucf.....	121
A.3.3	Arquivo package_main.vhd.....	121
A.3.4	Arquivo circuito_teste.vhd.....	122
A.3.5	Arquivo jk_borda_neg .vhd.....	123
A.3.6	Arquivo porta_nand.vhd .....	123
A.3.7	Arquivo porta_ne_3in.vhd.....	124
A.3.8	Arquivo porta_ou.vhd.....	124
<b>A.4</b>	<b>Código Fonte do Circuito 1 Modificado para o Cyclone II.....</b>	<b>124</b>
A.4.1	Arquivo main.ucf.....	124
<b>A.5</b>	<b>Código Fonte do Circuito 2 para o Spartan 3 .....</b>	<b>125</b>
A.5.1	Arquivo main.vhd.....	125
A.5.2	Arquivo main.ucf.....	126
A.5.3	Arquivo package_main.vhd.....	126
A.5.4	Arquivo gerador_sequencia.vhd .....	127
A.5.5	Arquivo arquivo_teste.vhd .....	128
A.5.6	Arquivo analisador_resposta.vhd .....	129
A.5.7	Arquivo flipflop_dbn.vhd .....	129
A.5.8	Arquivo flipflop_dbp.vhd .....	130
A.5.9	Arquivo porta_e.vhd .....	130
A.5.10	Arquivo porta_ou.vhd.....	131
A.5.11	Arquivo porta_inversora.vhd.....	131
<b>A.6</b>	<b>Código Fonte do Circuito 2 para o Cyclone II .....</b>	<b>131</b>
A.6.1	Arquivo main.ucf.....	131
A.6.2	Arquivo arquivo_teste.vhd .....	132
<b>A.7</b>	<b>Código Fonte do Oscilador em Anel para o SPARTAN 3.....</b>	<b>133</b>
A.7.1	Arquivo main.vhd.....	133
A.7.2	Arquivo main.ucf.....	133
A.7.3	Arquivo package_main.vhd.....	133
A.7.4	Arquivo inversores_serie.vhd.....	134
A.7.5	Arquivo porta_inversora.vhd.....	134
A.7.6	Arquivo porta_nand.vhd .....	135
<b>A.8</b>	<b>Código Fonte do Oscilador em Anel para o Cyclone II.....</b>	<b>135</b>
A.8.1	Arquivo main.ucf.....	135
A.8.2	Arquivo inversores_serie.vhd.....	135
<b>APÊNDICE B: SPARTAN 3 E CYCLONE II.....</b>		<b>137</b>
<b>APÊNDICE C: GRÁFICOS 3D COM OBSERVADOR POSICIONADO</b>		
	<b>EM OUTRO LOCAL .....</b>	<b>140</b>
C.1	Gráficos 3D dos Circuitos Implementados no Spartan 3.....	140
C.2	Gráficos 3D dos Circuitos Implementados no Cyclone II.....	143
<b>APÊNDICE D: ÁREAS DE RESTRIÇÃO .....</b>		<b>145</b>
D.1	Áreas de Restrição do Spartan 3 .....	145

<b>D.2 Áreas de Restrição do Cyclone II .....</b>	<b>148</b>
<b>APÊNDICE E: FÓRMULAS ESTATÍSTICAS .....</b>	<b>150</b>
<b>E.1 Média.....</b>	<b>150</b>
<b>E.2 Desvio-Padrão.....</b>	<b>150</b>
<b>E.3 Plano de Mínimos Quadrados.....</b>	<b>151</b>
<b>E.4 Média Móvel .....</b>	<b>151</b>

## LISTA DE ILUSTRAÇÕES

Figura 3.1 - Diagrama esquemático do circuito 1 .....	27
Figura 3.2 - Formas de onda do circuito 1 .....	28
Figura 3.3 - Diagrama esquemático do circuito 2 .....	29
Figura 3.4 - Exemplo de formas de onda do circuito 2 quando o período do <i>clock</i> é maior do que tempo gasto para o sinal percorrer todos os caminhos sobre teste .....	31
Figura 3.5 - Exemplo de formas de onda do circuito 2 quando o período do <i>clock</i> é menor do que tempo gasto para o sinal percorrer todos os caminhos sobre teste .....	32
Figura 3.6 - Diagrama esquemático do circuito 3 - oscilador em anel .....	33
Figura 4.1 - Placa Starter Spartan 3 da Digilent .....	35
Figura 4.2 - Placa Development and Education DE2 da Altera .....	36
Figura 5.1 - Diagrama esquemático do circuito 1 prototipado na placa Spartan-3 .....	41
Figura 5.2 - Vista da área utilizada no FPGA pelo circuito 1 colocado na posição 2. ....	42
Figura 5.3 - Vista do circuito 1 colocado na posição 2 do FPGA .....	42
Figura 5.4 - Número de ocorrências dos valores de cada área em relação ao número de ciclos do sinal de saída do circuito 1 no FPGA 2XC3S200 .....	44
Figura 5.5 - Número de ciclos do sinal de saída do circuito 1 associado as coordenadas do FPGA 2XC3S200 .....	45
Figura 5.6 - Média móvel dos dados do circuito 1 do FPGA 2XC3S200 .....	46
Figura 5.7 - Diferença entre o conjunto de dados e a média móvel do circuito 1 do FPGA 2XC3S200 .....	47
Figura 5.8 - Plano de mínimos quadrados dos dados do circuito 1 do FPGA 2XC3S200 .....	47
Figura 5.9 - Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 1 do FPGA 2XC3S200 .....	48
Figura 5.10 - Diagrama esquemático do circuito 1 modificado prototipado na placa Spartan-3 .....	49
Figura 5.11 - Formas de onda das saídas Q11(acima) e Q12 (abaixo) do circuito 1 modificado na segunda posição do FPGA XC3S200.....	50
Figura 5.12 - Formas de onda utilizadas para medir o tempo de atraso do flip-flop 12 do circuito 1 modificado implementado na segunda posição do FPGA XC3S200 .....	50
Figura 5.13 - Diagrama esquemático do circuito 1 prototipado na placa Cyclone II.....	53
Figura 5.14 - Vista da área utilizada no FPGA pelo circuito 1 colocado na posição 16. ....	53
Figura 5.15 - Número de ocorrências dos valores de cada área em relação ao número de ciclos do sinal de saída do circuito 1 no FPGA EP2C35F672.....	55
Figura 5.16 - Número de ciclos do sinal de saída do circuito 1 associado as coordenadas do FPGA EP2C35F672.....	57
Figura 5.17 - Média móvel dos dados do circuito 1 do FPGA EP2C35F672 .....	57
Figura 5.18 - Diferença entre o conjunto de dados e a média móvel do circuito 1 do FPGA EP2C35F672.....	58

Figura 5.19 - Plano de mínimos quadrados dos dados do circuito 1 do FPGA EP2C35F672 .....	59
Figura 5.20 - Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 1 do FPGA EP2C35F672 .....	60
Figura 5.21 - Diagrama esquemático do circuito 1 modificado prototipado na placa Cyclone II .....	61
Figura 5.22 - Diagrama esquemático do circuito 2 prototipado na placa Spartan-3 .....	63
Figura 5.23 - Forma de onda aplicada na entrada do circuito 2 implementado na área 2 do FPGA da placa Spartan 3.....	63
Figura 5.24 - Número de ocorrências dos valores médios medidos de cada área em relação à frequência máxima do sinal de <i>clock</i> do circuito 2 no FPGA 2XC3S200 .....	67
Figura 5.25 - Frequência do sinal de <i>clock</i> do circuito 2 associada as coordenadas do FPGA 2XC3S200.....	69
Figura 5.26 - Média móvel dos dados do circuito 2 do FPGA 2XC3S200 .....	70
Figura 5.27 - Diferença entre o conjunto de dados e a média móvel do circuito 2 do FPGA 2XC3S200 .....	71
Figura 5.28 - Plano de mínimos quadrados dos dados do circuito 2 do FPGA 2XC3S200 .....	72
Figura 5.29 - Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 2 do FPGA 2XC3S200 .....	72
Figura 5.30 - Diagrama esquemático do circuito 2 prototipado na placa Cyclone II.....	73
Figura 5.31 - Mudando o posicionamento dos caminhos sobre teste do circuito 2 da área 25 (rosa) para a área 26 (azul) no FPGA EP2C35F672.....	73
Figura 5.32 - Forma de onda aplicada na entrada do circuito 2 implementado no FPGA EP2C35F672.....	74
Figura 5.33 - Número de ocorrências dos valores médios de cada área em relação à frequência máxima do sinal de <i>clock</i> do circuito 2 no FPGA EP2C35F672.....	75
Figura 5.34 - Frequência do sinal de <i>clock</i> do circuito 2 associada as coordenadas do FPGA EP2C35F672.....	77
Figura 5.35 - Média móvel dos dados do circuito 2 do FPGA EP2C35F672 .....	78
Figura 5.36 - Diferença entre o conjunto de dados e a média móvel do circuito 2 do FPGA EP2C35F672 .....	79
Figura 5.37 - Plano de mínimos quadrados dos dados do circuito 2 do FPGA EP2C35F672 .....	80
Figura 5.38 - Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 2 do FPGA EP2C35F672.....	80
Figura 5.39 - Diagrama esquemático do oscilador em anel prototipado na placa Spartan-3 .....	81
Figura 5.40 - Vista da área disponível para o oscilador em anel colocado na posição 2 do FPGA XC3S200. ....	81
Figura 5.41 - Vista do oscilador em anel colocado na posição 2 do FPGA XC3S200. ....	82
Figura 5.42 - Forma de onda de saída do oscilador em anel colocado na primeira posição do FPGA XC3S200. ....	82
Figura 5.43 - Número de ocorrências dos valores de cada área em relação a frequência do sinal de saída do oscilador em anel no FPGA 2XC3S200 .....	84
Figura 5.44 - Frequência do sinal de saída do oscilador em anel associado às coordenadas do FPGA 2XC3S200.....	86
Figura 5.45 - Média móvel dos dados do oscilador em anel do FPGA 2XC3S200 .....	87

Figura 5.46 - Diferença entre o conjunto de dados e a média móvel do oscilador em anel do FPGA 2XC3S200.....	88
Figura 5.47 - Plano de mínimos quadrados dos dados do oscilador em anel do FPGA 2XC3S200.....	89
Figura 5.48 - Diferença entre o conjunto de dados e o plano de mínimos quadrados do oscilador em anel do FPGA 2XC3S200.....	89
Figura 5.49 - Diagrama esquemático do oscilador em anel prototipado na placa Cyclone II.....	90
Figura 5.50 - Forma de onda de saída do circuito oscilador em anel colocado na primeira posição do FPGA EP2C35F672.....	90
Figura 5.51 - Número de ocorrências dos valores das áreas em relação à frequência do sinal de saída do oscilador em anel no FPGA EP2C35F672.....	92
Figura 5.52 - Frequência do sinal de saída do oscilador em anel associado às coordenadas do FPGA EP2C35F672.....	94
Figura 5.53 - Média móvel dos dados do oscilador em anel do FPGA EP2C35F672.....	95
Figura 5.54 - Diferença entre o conjunto de dados e a média móvel do oscilador em anel do FPGA EP2C35F672.....	96
Figura 5.55 - Plano de mínimos quadrados dos dados do oscilador em anel do FPGA EP2C35F672.....	97
Figura 5.56 - Diferença entre o conjunto de dados e o plano de mínimos quadrados do oscilador em anel do FPGA EP2C35F672.....	97
Figura 5.57 - Gráficos comparando o valor médio e o desvio padrão de cada porta lógica do circuito 2 e do oscilador em anel no (a) Spartan 3 e (b) no Cyclone II.....	100
Figura 5.58 - Gráficos 3D do circuito 2 e oscilador em anel no Spartan 3.....	100
Figura 5.59 - Gráficos 3D do circuito 2 e oscilador em anel no Cyclone II.....	101
Figura B.1 - Arquitetura do Spartan 3 XC3S200FT256.....	137
Figura B.2 - Arquitetura do Cyclone II EP2C35F672.....	139
Figura C.1 - Gráfico opcional 1 da figura 5.5.....	140
Figura C.2 - Gráfico opcional 2 da figura 5.5.....	141
Figura C.3 - Gráfico opcional 1 da figura 5.25.....	141
Figura C.4 - Gráfico opcional 2 da figura 5.25.....	142
Figura C.5 - Gráfico opcional 1 da figura 5.44.....	142
Figura C.6 - Gráfico opcional 2 da figura 5.44.....	143
Figura C.7 - Gráfico opcional da figura 5.16.....	143
Figura C.8 - Gráfico opcional da figura 5.34.....	144
Figura C.9 - Gráfico opcional da figura 5.52.....	144
Figura D.1 - Áreas 1, 3, 5, 8, 10 e 12.....	145
Figura D.2 - Áreas 2, 4, 6, 7, 9 e 11.....	145
Figura D.3 - Áreas 13, 14, 15, 16, 17 e 18.....	146
Figura D.4 - Áreas 19, 22, 23, 26 e 27.....	146
Figura D.5 - Áreas 20, 21, 24, 25 e 28.....	147
Figura D.6 - Áreas 29, 30, 31, 32 e 33.....	147
Figura D.7 - Áreas 1, 3, 5, 7 e 9.....	148
Figura D.8 - Áreas 2, 4, 6, 8 e 10.....	148
Figura D.9 - Áreas 11, 12, 13, 14 e 15.....	148
Figura D.10 - Áreas 16, 19, 20, 23 e 24.....	149
Figura D.11 - Áreas 17, 18, 21, 22 e 25.....	149
Figura D.12 - Áreas 26, 27, 28, 29 e 30.....	149

## LISTA DE TABELAS

Tabela 4.1 - Áreas de restrição do circuito 1 no FPGA XC3S200.....	38
Tabela 4.2 - Áreas de restrição do circuito 1 no FPGA EP2C35F672 .....	39
Tabela 5.1 - Resultados do circuito 1 no FPGA 2XC3S200 .....	43
Tabela 5.2 - Resultados do circuito 1 modificado no FPGA 2XC3S200 .....	51
Tabela 5.3 - Resultados do circuito 1 no FPGA EP2C35F672 .....	54
Tabela 5.4 - Resultados do circuito 1 modificado no FPGA EP2C35F672 .....	62
Tabela 5.5 - Resultados das medidas efetuadas com o circuito 2 no FPGA 2XC3S200 .....	64
Tabela 5.6 - Resultados das medidas efetuadas com o circuito 2 no FPGA EP2C35F672 .....	75
Tabela 5.7 - Resultados do oscilador em anel no FPGA XC3S200 .....	83
Tabela 5.8 - Resultados do levantamento prático do oscilador em anel com 1350 inversores no FPGA EP2C35F672 .....	91
Tabela 5.9 - Resumo dos cálculos dos valores medidos .....	98
Tabela 5.10 - Resumo dos cálculos relacionados ao tempo de atraso de cada porta.....	99
Tabela 5.11 - Resumo das relações dos resíduo com os valores medidos no Spartan 3.....	99
Tabela 5.12 - Resumo das relações dos resíduo com os valores medidos no Cyclone II.....	99
Tabela 5.13 - Análise do tempo de atraso do gráfico 3D no Spartan 3 .....	101
Tabela 5.14 - Análise do tempo de atraso do gráfico 3D no Cyclone II .....	101
Tabela A.1 - Relação entre a pinagem do FPGA e os sinais de entrada e saída do circuito 1 no Cyclone II.....	117
Tabela A.2 - Relação entre a pinagem do FPGA e os sinais de entrada e saída do circuito 1 modificado no Cyclone II .....	125
Tabela A.3 - Relação entre a pinagem do FPGA e os sinais de entrada e saída do circuito 2 no Cyclone II.....	132
Tabela A.4 - Relação entre a pinagem do FPGA e os sinais de entrada e saída do circuito 4 no Cyclone II.....	135
Tabela B.1 - Resumo das características do Spartan 3 .....	137
Tabela B.2 - Resumo das características do Cyclone II .....	138

## LISTA DE ABREVIATURAS

3D	Três Dimensões
ASIC	Circuito Integrado de Aplicação Específica ( <i>Application-Specific Integrated Circuit</i> )
BIST	Auto-Teste Integrado ( <i>Built-In Self-Test</i> )
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CLB	Blocos Lógicos Programáveis ( <i>Configurable Logic Blocks</i> )
CMOS	Semicondutor de Óxido Metálico Complementar ( <i>Complementary Metal Oxide Semiconductor</i> )
D2D	<i>Die-to-Die</i>
DCM	<i>Digital Clock Manager</i>
DE2	Desenvolvimento e Educação 2 ( <i>Development and Education 2</i> )
FPGA	Matrizes de Portas Programáveis em Campo ( <i>Field Programmable Gate Array</i> )
HCI	Injeção de Portadores Quentes ( <i>Hot Carrier Injection</i> )
IFSUL	Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense
I/O	Entrada/Saída
IOB	Bloco de Entrada/Saída
IOE	Elementos de Entrada/Saída
LAB	Bloco de Matriz Lógica ( <i>Logic Array Block</i> )
LE	Elemento Lógico ( <i>Logic Element</i> )
LUT	<i>Look-Up Table</i>
LVDS	Sinalização Diferencial de Baixa Tensão ( <i>Low-Voltage Differential Signaling</i> )
NBTI	Instabilidade por Temperatura e Tensão Negativa ( <i>Negative Bias Temperature Instability</i> )
PLB	Blocos Lógicos Programáveis
PLL	<i>Phase-Locked Loops</i>
PPGEE	Programa de Pós-Graduação em Engenharia Elétrica
PUT	Caminhos Sobre Teste ( <i>Paths Under Test</i> )
RAM	Memória de Acesso Aleatório ( <i>Random Access Memory</i> )
RDF	Flutuação Aleatória do Número de Dopantes ( <i>Random Dopant Fluctuation</i> )

VHDL Linguagem de Descrição de Circuito Integrado de Velocidade Muito Alta  
(*Very High Speed Integrated Circuit Hardware Description Language*)

Vth Tensão de limiar

WID Intra-Pastilha (*Within-Die*)

## 1 INTRODUÇÃO

Nos últimos 20 anos tem aumentado o uso das matrizes de portas programáveis em campo (*Field Programmable Gate Arrays* - FPGA) na produção de circuitos digitais (CARRO, 2001; COSTA, 2009; HARRIS, 2001; MENON, 2006; SMITH, 2007; SMITH, 2009). Esse aumento do uso do FPGA é devido à prototipagem rápida (CARRO, 2001; MENON, 2006; SMITH, 2007; SMITH, 2009), à produção de circuitos que acompanham o desenvolvimento de novas tecnologias (COSTA, 2009), aos custos menores (CARRO, 2001; HARRIS, 2001) e ao atendimento das demandas dos usuários (CARRO, 2001; COSTA, 2009).

A performance dos circuitos digitais produzidos com FPGA depende de vários parâmetros, tais como o atraso de propagação e a potência. No entanto, esses parâmetros estão sujeitos à variabilidade em decorrência de fatores físicos, ambientais e temporais. Além disso, o aumento da escala de integração da tecnologia do semicondutor de óxido metálico complementar (CMOS - *Complementary Metal Oxide Semiconductor*) faz com que esses parâmetros fiquem mais sujeitos à variabilidade que ocorre dentro do *chip* devido ao processo de fabricação (ABRAMOVICI, 2002; NEUBERGER, 2009; NIKOLIC, 2006; SEDCOLE, 2006).

O atraso de propagação também pode ser alterado durante a escolha do roteamento global ou local do circuito no *chip*, já que os recursos de roteamento ocupam, aproximadamente, 80% dos transistores do FPGA e uma área maior que os blocos lógicos (SMITH, 2009).

Portanto, as variações decorrentes do roteamento e da variabilidade dos parâmetros dos circuitos digitais podem provocar falhas no funcionamento dos sistemas de alta

performance. E como é impossível para o fabricante de FPGA testar todos os circuitos que são implementados em cada dispositivo, o usuário pode aproveitar a reconfigurabilidade dos FPGA para medir e caracterizar a variação da performance de cada dispositivo usando uma estrutura de auto-teste integrado (BIST - *Built-In Self-Test*). Sendo assim, os testes de falhas devidos ao atraso de propagação em FPGA têm aumentado de importância (HARRIS, 2001; MENON, 2006).

Em consequência disso, esta dissertação propõe uma metodologia de análise da variabilidade do tempo de atraso de propagação de um circuito posicionado dentro de várias áreas de um FPGA. Assim, a variabilidade estará relacionada às variações dos parâmetros devido aos fatores físicos e às variações decorrentes do roteamento que ocorrem dentro de um *chip* (MENON, 2006). A escolha do roteamento ficará a critério da ferramenta utilizada.

Inicialmente, este estudo propõe-se à realização de uma revisão bibliográfica com a apresentação de conceitos relevantes sobre os fatores que provocam a variabilidade do FPGA, além da busca de circuitos adequados para a caracterização da variabilidade dentro de um mesmo *chip*.

Embasado na revisão bibliográfica, foram escolhidos três circuitos de teste para a realização da caracterização proposta.

No primeiro circuito foi usada uma técnica que avalia a diferença dos tempos de atraso de propagação de dois circuitos combinacionais iguais. Nesse circuito, o número de pulsos do sinal de saída é diretamente proporcional à diferença dos tempos de atraso dos dois circuitos combinacionais sobre teste. Esse circuito pode ser usado para testar um número maior de circuitos iguais ao mesmo tempo.

No segundo circuito foi empregada uma técnica que avalia os tempos de atraso de propagação de dois circuitos combinacionais iguais. O sinal de saída terá nível 1 quando o sinal atravessar os dois caminhos sobre teste e uma porta E antes do *clock* completar um

pulso. Esse circuito pode ser usado para testar um circuito ou qualquer número de circuitos iguais ao mesmo tempo.

No terceiro circuito foi usado o oscilador em anel. Nesse circuito, o valor da frequência de saída é usado para avaliar a variabilidade do atraso de propagação dentro do FPGA.

Todos os circuitos de teste foram implementados, utilizando-se a linguagem de descrição de hardware de circuito integrado de velocidade muito alta (VHDL - *Very High Speed Integrated Circuit Hardware Description Language*), no FPGA Spartan 3 XC3S200-FT256 da placa Starter Spartan 3 da Digilent e no FPGA Cyclone II EP2C35F672C6 da placa Development and Education DE2 da Altera. No Spartan 3, os circuitos foram implementados em 33 posições diferentes e no Cyclone II os circuitos foram implementados em 30 posições diferentes. Como na maioria dos métodos propostos para testar FPGA, foi utilizada a estrutura BIST.

Devido aos kits serem de companhias diferentes, optou-se por usar os softwares de que são proprietários para se fazer a implementação dos circuitos nos FPGA. Portanto, os softwares usados foram o Xilinx ISE Design Suite 10.1 e o Quartus II Design Software Version 7.2.

Os equipamentos utilizados para executar os testes foram o gerador de função BK PRECISION 4086 80MHz Arbitrary/Function Generator with Counter e o osciloscópio Agilent Technologies DSO 3062A Digital Storage Oscilloscope 60MHz 1GSa/s. Também foi utilizado um termômetro com escala de  $-10^{\circ}\text{C}$  a  $+150^{\circ}\text{C}$  e divisão de  $1^{\circ}\text{C}$  para medir a temperatura ambiente.

Os ambientes escolhidos para a realização dos ensaios foram os laboratórios do Curso Técnico em Telecomunicações do Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense do Campus Pelotas.

Como possíveis fontes de erros dos resultados dos ensaios pode-se citar: o uso de um osciloscópio com uma faixa característica do sistema horizontal de 2 ns/div a 50 s/div para medir o tempo de atraso dos flip-flops do circuito 1 modificado; a falta de terminais de entrada e saída de alta frequência no kit Spartam 3; a falta de terminais de saída de alta frequência no kit DE2; a frequência gerada no oscilador do circuito 1 ultrapassar a frequência máxima recomendada pelo fabricante para os kits; o tamanho das áreas onde serão colocados os circuitos; a variação da temperatura ambiente e o valor da tensão aplicada.

A partir dos resultados obtidos, pôde-se concluir que tanto a metodologia adotada quanto a identificação da variabilidade do tempo de atraso de propagação de um circuito posicionado dentro de várias áreas de um FPGA mostram a existência de variações *within-die* em FPGA e que essa variabilidade pode provocar falhas no funcionamento de um circuito digital. Por isso, todos os sistemas de alta performance feitos com FPGA, que pretendem atingir o máximo desempenho possível para uma dada tecnologia, antes de serem usados, devem ter o seu funcionamento testado devido à variabilidade dos parâmetros.

Esta dissertação encontra-se dividida em 6 capítulos, conforme descrito a seguir:

No primeiro capítulo é apresentada a introdução, onde são abordadas as vantagens do FPGA; as causas da variabilidade do atraso de propagação e das falhas no funcionamento de circuitos; os objetivos, circuitos e FPGA empregados; a linguagem de descrição de hardware; os softwares usados na implementação dos circuitos; os equipamentos necessários; o local dos testes, as fontes de erros, o resumo da conclusão e a estrutura do trabalho.

No capítulo 2, resume-se o princípio de funcionamento de alguns circuitos, encontrados durante a revisão bibliográfica, que realizam testes de falhas em FPGA devido ao tempo de atraso de propagação.

No capítulo 3, encontra-se a fundamentação teórica. Descrita, de forma sucinta, as causas da variabilidade do atraso de propagação e das falhas no funcionamento de circuitos,

algumas classificações da variabilidade que ocorre nos FPGA e os circuitos utilizados. Nestes, são abordados o diagrama esquemático, o princípio de funcionamento, o diagrama de tempo e as equações associadas ao tempo de propagação de cada porta lógica.

No capítulo 4, apresenta-se a metodologia empregada na dissertação. Neste item, são abordadas as etapas percorridas, FPGA, linguagem de descrição de hardware, softwares e arquivos usados na implementação dos circuitos, configuração de software, equipamentos utilizados, local dos testes, motivos que levaram a escolha dos circuitos e das áreas dos circuitos sobre testes, áreas de restrição, e a metodologia da análise.

No capítulo 5, encontram-se os resultados e a análise. Esse capítulo mostra o circuito adaptado ao kits, a temperatura ambiente de cada ensaio, a rotina utilizada em cada ensaio, os dados obtidos nos ensaios, os valores médio e desvio padrão resultante dos dados obtidos, a análise dos gráficos de três dimensões (3D) individuais e a discussão dos resultados.

Por fim, no capítulo 6, são discutidas as conclusões e as perspectivas obtidas com este trabalho.

## 2 LITERATURA EM CARACTERIZAÇÃO DE FPGA

Para a elaboração deste trabalho foram revisados vários artigos cuja proposta envolve a análise de falhas em FPGA devido ao tempo de atraso de propagação e à criação de circuitos para testes de falhas em FPGA provocadas pelo tempo de atraso de propagação.

Existem vários circuitos para testar o FPGA em relação ao tempo de atraso de propagação (MENON, 2006). A maioria dos métodos propostos para testar FPGA utiliza uma estrutura denominada de BIST, ou seja, o circuito de teste é colocado dentro do FPGA em áreas não usadas pelo caminho sobre teste (GONSALES 2002; HARRIS, 2001; MENON, 2006; SMITH, 2007, SMITH, 2009). Aqui são brevemente comentados, dentre os trabalhos encontrados na literatura, os mais relevantes no contexto desta dissertação.

Smith utilizou um circuito para teste de atraso de propagação de alta-resolução para a rede de conexão do FPGA. No Xilinx Virtex-4, esse teste detectou defeitos de 78 ps usando o circuito de linha de atraso IDELAY que está disponível nesse dispositivo.

Menon empregou um circuito para teste de atraso em caminhos de *look-up-table* (LUT) de FPGA usando um gerador de seqüência, um analisador de resposta e um circuito de controle.

Abramovici propôs um circuito para verificar falhas de atraso de caminhos iguais de modo indireto. Nesse circuito, uma borda positiva ou negativa é aplicada simultaneamente na entrada de todos os caminhos sobre teste. Na saída dos caminhos sobre teste há um oscilador em que o número de pulsos é diretamente proporcional à diferença de tempo entre o caminho mais lento e mais rápido. A falha é constatada quando o número de pulsos do sinal de saída ultrapassa a 21 ciclos.

Sedcole sugeriu o uso de uma matriz de 34 linhas e 26 colunas de osciladores em anel, implementada em 18 dispositivos Cyclone II, para estudar o comportamento e a variabilidade do atraso de propagação dentro de um FPGA de 90nm.

Wong criou um teste de auto-caracterização do tempo de atraso de um circuito combinacional no FPGA. Nesse teste, a frequência do *clock* é elevada até que uma falha seja detectada no circuito combinacional.

O estudo desta literatura serviu como base para o desenvolvimento da metodologia empregada no desenvolvimento desta dissertação, conforme descrito nos próximos capítulos.

## 3 FUNDAMENTAÇÃO TEÓRICA

### 3.1 Variabilidade

Como nos circuitos integrado de aplicação específica (*ASIC - Application-Specific Integrated Circuit*), os FPGA estão sujeitos às variações no funcionamento dos transistores das funções lógicas e das chaves de interconexão programáveis (LIN, 2006). O desvio destes valores nominais dos parâmetros dos dispositivos origina-se por consequência de uma das três causas: fatores físicos, ambientais e temporais (NEUBERGER, 2007). Os fatores físicos referem-se às variações que são introduzidas como resultado de alteração no material devido aos materiais e ao processo de fabricação (MUKHERJEE, 2006). As variações devido ao processo de fabricação são flutuações observadas nos valores dos parâmetros do processo após a fabricação do dispositivo e são devidas a falta de controle sobre o processo de fabricação (NEUBERGER, 2007). Como exemplo de fatores físicos pode-se citar a variabilidade que ocorre dentro do *chip* devido ao processo de fabricação e a máscara. Os fatores ambientais consistem em variações que ocorrem como resultado de alterações no ambiente em que o *chip* se encontra durante o seu funcionamento. (MUKHERJEE, 2006; NEUBERGER, 2007). Como exemplo de variações decorrente do ambiente pode-se citar as variações de temperatura e as variações da tensão de alimentação. As variações temporais são aquelas que fazem o *chip* comportar-se de maneira diferente em tempos diferentes (NEUBERGER, 2007). Em relação aos fatores temporais pode-se citar os transientes causados por radiações ionizante (GHISSONI, 2009) e os efeitos do envelhecimento, tais como a instabilidade por temperatura e tensão negativa (NBTI - *Negative Bias Temperature Instability*)( KUFLUOGLU, 2004; NEUBERGER, 2007) e a injeção à quente de portadores (HCI - *Hot Carrier Injection*)(KUFLUOGLU, 2004) que causam a degradação do desempenho do transistor CMOS ao longo da vida útil.

Como a densidade de transistores da tecnologia CMOS continua aumentando, os efeitos induzidos pelo processo de fabricação estão levando ao aumento das variações dos parâmetros dos circuitos (ABRAMOVICI, 2002; DAS, 2009; NEUBERGER, 2009; NIKOLIC, 2006; SEDCOLE, 2009). Esse aumento na variabilidade dos parâmetros dos circuitos dentro de um *chip* tem sido um dos principais obstáculos para o aumento da escala de integração (NIKOLIC, 2006). E tem afetado parâmetros importantes, como o atraso e a potência, os quais podem alterar a própria performance do circuito e produzir falhas no seu funcionamento.

### 3.1.1 Classificação da variabilidade de processo

As variações dos parâmetros de processo podem ser classificadas em *within-die* e *die-to-die* (LIN, 2006; NEUBERGER, 2007).

As variações *within-die* (WID ou *intra-die* ou local) afetam transistores diferentes dentro de um *chip* de forma diferente. Como exemplo, tem-se as variações aleatórias que afetam todos os *gates* de um *chip* de forma diferente. (NEUBERGER, 2007). Lin denominou de espacial ou regional as variações que afetam áreas geográficas de um *chip*. Como exemplo, pode-se citar que os *gates* de uma pequena região estão mais correlacionados quando comparado aos *gates* de uma região maior.

As variações *die-to-die* (D2D ou *inter-die* ou global) afetam todas as áreas dentro de um *chip* da mesma forma e, entre *chips*, de forma diferente (NEUBERGER, 2007). Como exemplo, tem-se as variações que afetam todos os *gates* dentro de um *chip* de forma idêntica e que, entre *chips*, são diferentes.

Essas variações ainda podem ser divididas em uma componente aleatória e em uma componente sistemática (MATSUMOTO, 2007; NEUBERGER, 2007; SEDCOLE, 2009). As variações sistemáticas estão relacionados, principalmente, aos padrões do circuito e devem ser

tratadas como problemas de leiaute, projeto da máscara e imperfeições nos equipamentos usados para a fabricação do circuito integrado (MATSUMOTO, 2007). Por outro lado, as variações aleatórias são de natureza estocástica, uma certa parte do que são intrinsecamente incontroláveis na fabricação (MATSUMOTO, 2007). Um exemplo representativo dessas variações aleatórias intrínsecas é a variação na tensão de limiar de um transistor ( $V_{th}$ ), causada por fenômenos como flutuação aleatória do número de dopantes (RDF - *Random Dopant Fluctuations*) (DAS, 2009; MAHMOODI, 2005). Atualmente, as variações sistemáticas têm um impacto maior que as variações aleatórias na variabilidade dos parâmetros, mas o impacto das variações aleatórias intrínsecas devem aumentar no futuro (MATSUMOTO, 2007).

A partir dos conceitos acima, pode-se descrever o tempo de propagação de uma porta lógica como (DAS, 2009):

$$t_p = t_{p0} + \Delta t_{pg} + \Delta t_{pls} + \Delta t_{plr} \quad (3.1)$$

Onde:

$t_{p0}$  é o tempo de propagação nominal de cada porta lógica.

$\Delta t_{pg}$  é a variação global do tempo de propagação de cada porta lógica.

$\Delta t_{pls}$  é a variação da correlação espacial local do tempo de propagação de cada porta lógica.

$\Delta t_{plr}$  é a variação aleatória local do tempo de propagação de cada porta lógica.

Para uma área pequena do *chip* com correlação espacial, a variação global e a variação da correlação espacial local são iguais para todas as portas dessa região, sendo que a variação aleatória local é um valor individual das portas dessa região. Os circuitos desse trabalho não dão informações da variação aleatória local de uma porta lógica, mas do tempo de propagação de cada porta lógica com todas as variações incluídas.

### 3.1.2 Variabilidade decorrente do roteamento

Na estrutura simplificada do FPGA, o atraso de propagação ao longo de um caminho acumula o atraso de todos seus blocos lógicos, segmentos de trilhas e chaves de interconexão programáveis (ABRAMOVICI, 2002). Por isso, a rede de conexão formada pelos segmentos de trilhas conectados por chaves de interconexão programáveis são considerados recursos de roteamento (COSTA, 2009). Os recursos de roteamento ocupam aproximadamente 80% dos transistores do FPGA e ocupam uma área maior que os blocos lógicos (SMITH, 2007; SMITH, 2009). A escolha entre um roteamento global ou local de um circuito pode alterar o atraso de propagação ao longo de um caminho. Os recursos de roteamento globais são considerados aqueles que conectam blocos lógicos programáveis (PLB) não adjacentes e os recursos de roteamento locais são considerados aqueles que conectam PLB adjacentes (ABRAMOVICI, 2002).

### 3.2 Circuitos Utilizados

Após o estudo do estado da arte, foram escolhidas três técnicas de medição adequadas para a caracterização proposta. A escolha de três técnicas foi visando a uma maior credibilidade aos resultados. Nos três circuitos de teste, como na maioria dos métodos propostos para testar FPGA, foi utilizada a estrutura BIST.

No primeiro circuito foi usada uma técnica que avalia a diferença dos tempos de atraso de propagação de dois circuitos combinacionais iguais. Nesse circuito, o número de pulsos do sinal de saída é diretamente proporcional à diferença dos tempos de atraso dos dois circuitos combinacionais sobre teste. Esse circuito pode ser usado para testar um número maior de circuitos iguais ao mesmo tempo.

No segundo circuito foi empregada uma técnica que avalia os tempos de atraso de propagação de dois circuitos combinacionais iguais. O sinal *output* só tem nível 1 quando o

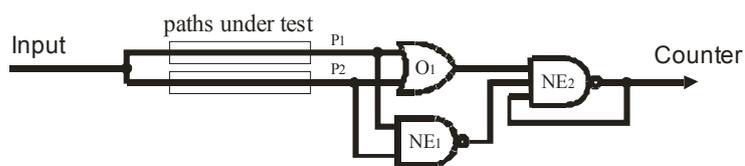
sinal atravessar os dois caminhos sobre teste e uma porta E antes do *clock* completar um pulso. Esse circuito pode ser usado para testar um circuito ou qualquer número de circuitos iguais ao mesmo tempo.

No terceiro circuito foi utilizado o oscilador em anel. Neste circuito, o valor da frequência de saída é usado para avaliar a variabilidade do atraso de propagação dentro do FPGA.

### 3.2.1 Circuito 1

A figura 3.1 mostra o diagrama esquemático do circuito 1 com ilustração de somente dois caminhos sobre teste (*Paths Under Test - PUT*). Cada caminho sobre teste representa um circuito a ser usado na medição do tempo de atraso de propagação. A única diferença entre os PUT deve ser o posicionamento dentro do FPGA.

Esse circuito foi originalmente usado no trabalho “BIST – *Based Delay-Fault Testing in FPGAs*” para comparar os atrasos de propagação dos PUT, tendo capacidade de testar vários PUT iguais ao mesmo tempo (ABRAMOVICI, 2002).



**Figura 3.1 - Diagrama esquemático do circuito 1.**

Nesse circuito existem dois sinais: o sinal de entrada *input* e o sinal de saída *counter*.

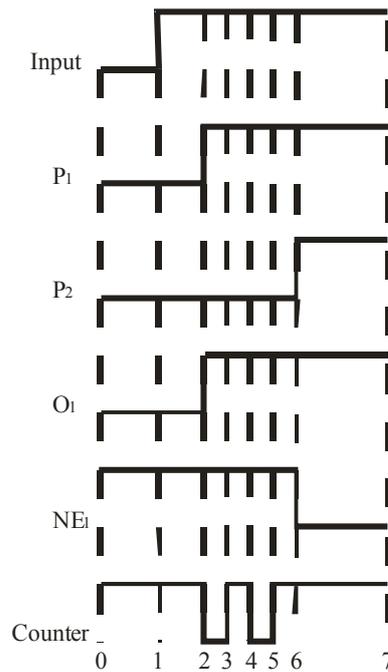
Se na entrada *input* do circuito 1 for aplicado um sinal de nível 0, tem-se nas entradas das portas  $O_1$  e  $NE_1$  um nível 0, na saída da porta  $O_1$  um nível 0, na saída da porta  $NE_1$  um nível 1 e na saída da porta  $NE_2$  um nível 1 (figura 3.2).

Se na entrada *input* no circuito 1 for aplicado um sinal que mude do nível 0 para o nível 1, tem-se na saída dos caminhos sobre teste a mesma transição com um atraso de propagação entre os caminhos. A porta  $O_1$  muda do nível 0 para o nível 1 quando ocorrer a

primeira mudança nas saídas dos PUT. A porta  $NE_1$  muda do nível 1 para o nível 0 quando ocorrer a última mudança nas saídas dos PUT. Essa diferença no tempo de chegada dos sinais, devido ao atraso de propagação nos caminhos sobre teste, faz com que a porta  $NE_2$  oscile (figura 3.2). O tempo de oscilação da porta  $NE_2$  é diretamente proporcional à diferença de tempo entre as mudanças de nível das portas  $NE_1$  e  $O_1$ . O circuito contador deve contar o número de pulsos apresentados na saída da porta  $NE_2$ .

Um caminho sobre teste pode ter atrasos diferentes para transições na borda de subida (0/1) e de descida(1/0).

É interessante observar que, diferentemente de outros circuitos de teste de atraso de propagação, esse circuito não precisa de um relógio.



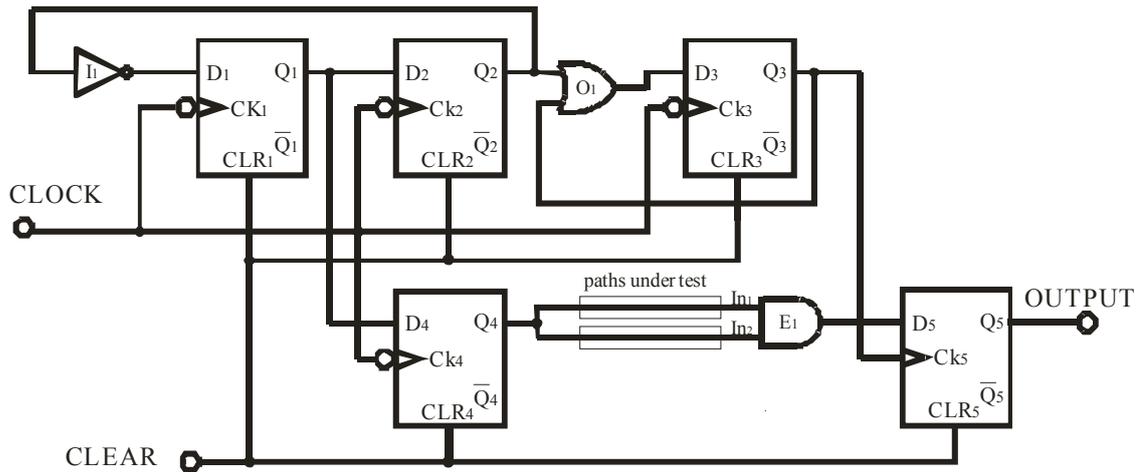
**Figura 3.2 – Formas de onda do circuito 1.**

### 3.2.2 Circuito 2

A figura 3.3 mostra o diagrama esquemático do circuito 2 com ilustração de somente dois caminhos sobre teste, mas esse circuito tem capacidade de testar um circuito ou qualquer

número de circuitos iguais ao mesmo tempo. A única diferença entre os PUT deve ser o posicionamento dentro do FPGA.

Quando for usado somente um caminho sobre teste, o circuito deve ser modificado. A porta E deve ser desligada e a saída do circuito a ser testado deve ser ligado a entradas D<sub>5</sub>. Também deve ser retirado o número 1 das fórmulas 3.3 e 3.4.



**Figura 3.3 - Diagrama esquemático do circuito 2.**

Nesse circuito existem três sinais: os sinais de entrada *clock* e *clear* e o sinal de saída *output*.

Quando a entrada *clear* do circuito 2 é colocada no nível 1, as saídas Q dos flip-flops vão para o nível 0 e o inversor I<sub>1</sub> terá sua saída no nível 1. Na sequência, o *clear* é colocado no nível 0 onde permanece até o final de cada medida. Na primeira borda negativa do *clock*, após o *clear* ter passado para o nível 0, a saída Q<sub>1</sub> vai para o nível 1. Na segunda borda negativa do *clock*, a saída Q<sub>1</sub> permanece em 1 e as saídas Q<sub>2</sub> e Q<sub>4</sub> passam para o nível 1. Nesse momento, um pulso positivo é aplicado nos caminhos a serem testados e a saída do inversor vai para o nível 0. Na terceira borda negativa do *clock*, a saída Q<sub>3</sub> passa para o nível 1 e a saída Q<sub>1</sub> volta para o nível 0. Com a saída Q<sub>3</sub> passando para o nível 1, a saída Q<sub>5</sub> mudará para o nível 1 se, e somente se, o pulso positivo aplicado nos caminhos a serem testados percorreram os mesmos e produziram na saída da porta E<sub>1</sub> o nível 1 (figura 3.4). Caso contrário, a saída Q<sub>5</sub>

permanece no nível 0 (figura 3.5). Portanto, o sinal *output* só tem nível 1 quando a mudança do sinal de  $Q_4$  do nível 0 para o nível 1 atravessar os caminhos sobre teste e a porta  $E_1$  antes do *clock* completar um pulso.

Conhecendo-se a frequência do *clock* ( $f$ ) quando o terminal *output* passa para o nível 1, é possível calcular o maior tempo de atraso de propagação dos circuitos sobre teste ( $t$ ).

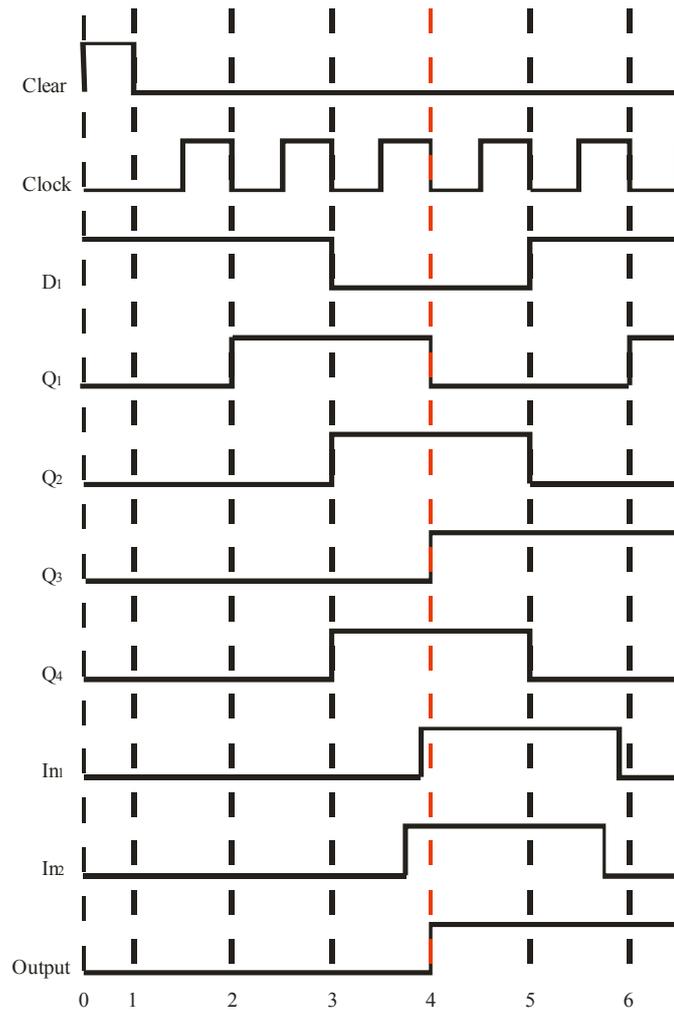
$$t = \frac{1}{f} \quad (3.2)$$

Considerando que o circuito sobre teste é composto por  $n$  inversores em série. Reescrevendo a equação acima em relação ao tempo de propagação de cada porta lógica ( $t_p$ ), tem-se:

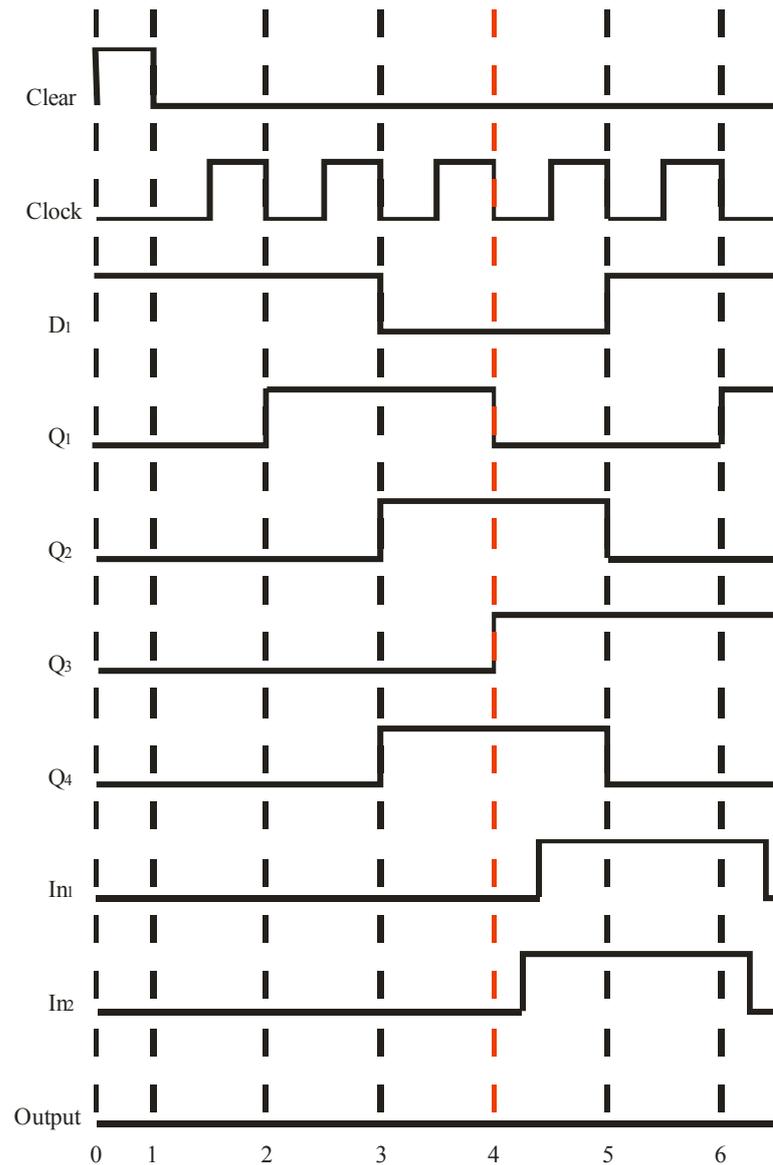
$$t = (n + 1) \cdot t_p \quad (3.3)$$

Em relação à frequência de oscilação do circuito, tem-se:

$$t_p = \frac{1}{(n+1) \cdot f} \quad (3.4)$$



**Figura 3.4 – Exemplo de formas de onda do circuito 2 quando o período do *clock* é maior do que tempo gasto para o sinal percorrer todos os caminhos sobre teste.**

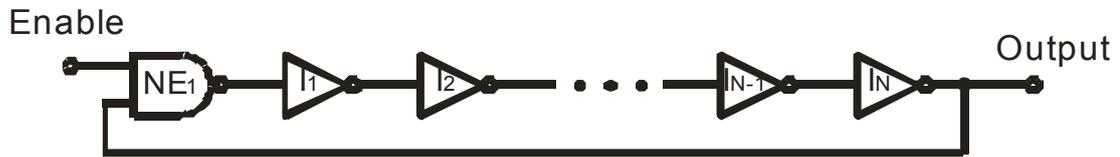


**Figura 3.5 – Exemplo de formas de onda do circuito 2 quando o período do *clock* é menor do que tempo gasto para o sinal percorrer todos os caminhos sobre teste.**

### 3.2.3 Circuito 3 – Oscilador em anel

Esse é um circuito usado freqüentemente (figura 3.6) para demonstrar uma nova tecnologia de hardware e pode ser encontrado em vários trabalhos (DAS, 2009; NIKOLIC, 2006; SEDCOLE, 2009). Em relação à medição da variação do atraso de propagação devido às variações no processo, esse circuito tem sido usado porque é de fácil implementação e por apresentar alta sensibilidade às variações dos parâmetros do processo (DAS, 2009). É composto por um número par de portas inversoras ligadas em série e por uma porta NAND.

Existem dois sinais principais: o sinal de entrada chamado de *enable* e o sinal de saída chamado de *output*. Se o *enable* estiver no nível lógico 1, o sinal de saída irá oscilar.



**Figura 3.6 - Diagrama esquemático do circuito 3 - oscilador em anel.**

Conhecendo-se a frequência de oscilação em *output* ( $f$ ), é possível calcular o tempo de atraso de propagação do circuito ( $t$ ).

$$t = \frac{1}{f} \quad (3.5)$$

Para completar um ciclo de oscilação no oscilador em anel, é necessário que cada porta lógica troque seu estado duas vezes (SEDRA, 2000). Então, o período do tempo de atraso de propagação ( $t$ ) do circuito fica:

$$t = 2 \cdot (n + 1) \cdot t_p \quad (3.6)$$

onde  $n$  é o número de inversores utilizados no circuito e  $t_p$  é o tempo de propagação de cada porta lógica. Ou em relação à frequência de oscilação do circuito, tem-se:

$$t_p = \frac{1}{2 \cdot (n+1) \cdot f} \quad (3.7)$$

## 4 METODOLOGIA

### 4.1 Introdução

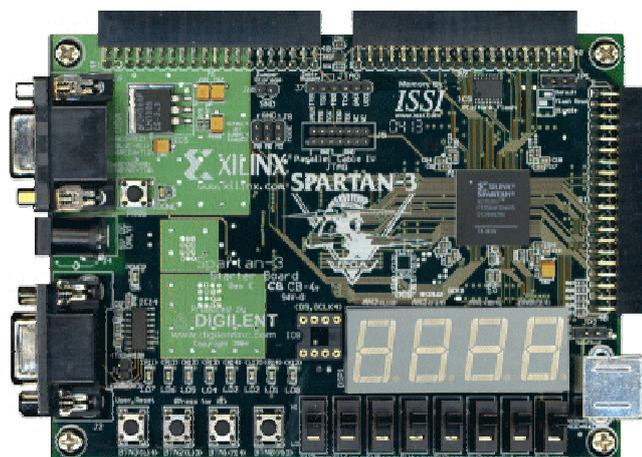
Para avaliar a existência da variabilidade do tempo de atraso de propagação de um circuito implementado em várias posições do FPGA foi necessário percorrer várias etapas.

Na primeira etapa foi feito um estudo do estado da arte na busca de circuitos adequados para a caracterização proposta. Nesses artigos foram encontrados vários circuitos que poderiam ser usados para fazer o trabalho. Mas visando dar maior credibilidade aos seus resultados, foram escolhidos três circuitos para avaliar a variabilidade do tempo de atraso de propagação de um circuito implementado em várias posições do FPGA. O circuito 1 foi escolhido por ser um circuito simples de montar, de fácil compreensão e não depender de um sinal de *clock*. O circuito 2 foi escolhido por ser um circuito que avalia o tempo de atraso de propagação de dois circuitos iguais no tempo de um ciclo de *clock* automaticamente. E o circuito 3, o oscilador em anel, por ser um circuito usado freqüentemente para demonstrar uma nova tecnologia de hardware.

Após a escolha dos circuitos, foi feita a escolha dos FPGA. Foram escolhidos o FPGA Xilinx Spartan 3 XC3S200-FT256 da placa Starter Spartan 3 da Digilent (ver figura 4.1) e o FPGA Cyclone II EP2C35F672C6 da placa Development and Education DE2 da Altera (ver figura 4.2).

Como os kits são de companhias diferentes, optou-se por usar os softwares de que são proprietários para se fazer a implementação dos circuitos nos FPGA. Portanto, os softwares usados foram o Xilinx ISE Design Suite 10.1 para implementar os circuitos no XC3S200-FT256 e o Quartus II Design Software Version 7.2 para implementar os circuitos no EP2C35F672C6. O Quartus II Version 7.2 foi usado devido a ser o único software da Altera com licença para implementar os circuitos no FPGA EP2C35F672C6. Estes softwares

estavam instalados em computadores diferentes, mas contendo as mesmas características, ou seja: processador INTEL Pentium Dual Core E2180 2.00 GHz com memória física total de 1 GB e Sistema Operacional de 32 Bits Microsoft Windows XP Professional com Service Pack 2.



**Figura 4.1 - Placa Starter Spartan 3 da Digilent.**

Fonte: DIGILENT, [2009?].

O software Notepad++ foi usado para produzir os arquivos de entrada (\*.vhd). O software Matlab version 6 release 12 foi empregado para obter as constantes  $a_0$ ,  $a_1$  e  $a_2$  de cada plano de mínimos quadrados construído e construir os histogramas e gráficos 3D. O software Microsoft Office Excel 2007 foi aproveitado para realizar todos os cálculos restantes dessa dissertação. E o GraphPad Prism foi aplicado para fazer o teste estatístico de hipótese para médias t-student e criar os gráficos que comparam o valor médio e o desvio padrão de cada porta lógica do circuito 2 e do oscilador em anel.

Com a seleção dos circuitos e dos kits, foi possível escolher os equipamentos, cabos, fios e conectores necessários para realizar os testes. Os equipamentos utilizados para executar os testes foram o gerador de função BK PRECISION 4086 80MHz Arbitrary/Function Generator with Counter e o osciloscópio Agilent Technologies DSO 3062A Digital Storage Oscilloscope 60MHz 1GSa/s.



**Figura 4.2 - Placa Development and Education DE2 da Altera.**

Fonte: ALTERA CORPORATION, [2009?].

Também foi usado um termômetro com escala de  $-10^{\circ}\text{C}$  a  $+150^{\circ}\text{C}$  e divisão de  $1^{\circ}\text{C}$  da marca Incoterm para medir a temperatura ambiente.

Os ambientes escolhidos para a realização dos ensaios foram os laboratórios 2 e 5 do Curso Técnico em Telecomunicações do Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense de Pelotas.

## **4.2 Procedimento Experimental**

Para verificar a variabilidade do tempo de atraso de propagação de um circuito implementado em várias posições do FPGA, foi usada a linguagem de descrição de hardware VHDL para fazer a descrição estrutural e comportamental dos circuitos escolhidos durante o estudo do estado da arte. O apêndice A contém os códigos dos projetos dos circuitos em VHDL com a correspondente descrição das chaves, pinos e leds utilizados. As chaves, pinos e leds foram selecionados a partir dos manuais dos fabricantes (ALTERA CORPORATION, 2006; XILINX, 2005).

Os diagramas esquemáticos dos circuitos que precisaram ser adaptados aos kits estão mostrados no item 5 “Resultados e Análises”.

As áreas onde foram colocados os circuitos sobre teste estão representadas nas tabelas 4.1 e 4.2, na forma de coordenadas, ou no apêndice D, na forma de desenho. As características de cada FPGA foram obtidas a partir dos manuais dos fabricantes (ALTERA CORPORATION, 2008; XILINX, 2009b XILINX, 2009c). Para implementar o circuito 1 ou o circuito 2 ou o oscilador em anel em uma área do FPGA XC3S200, deve-se utilizar a segunda e terceira coluna da tabela 4.1. Se for no FPGA EP2C35F672 devemos utilizar a segunda, terceira e quarta colunas da tabela 4.2. Para implementar o circuito 1 modificado em uma área do FPGA XC3S200, deve-se utilizar a quarta e quinta coluna da tabela 4.1. Se for no FPGA EP2C35F672, deve-se utilizar a quinta, sexta e sétima colunas da tabela 4.2.

Para obter uma amostra de tamanho grande, procurou-se repartir o FPGA em, no mínimo, 30 áreas (SPIEGEL, 1994). Ao mesmo tempo, a área escolhida deve ser apropriada à implementação de circuitos sobre teste que produzem um tempo atraso de propagação possível de ser medido com os instrumentos disponíveis.

Para cada área foram realizadas 10 medidas por circuito. No item 5 “Resultados e Análises” as tabelas com os resultados só indicam as 10 medidas quando estas são diferentes.

Antes do processo de síntese no software Xilinx ISE Design Suite 10.1 sempre foi alterada a propriedade “*Keep Hierarchy*” da *Synthesize - XST* para “Yes”. Com essa mudança, foi evitado que a ferramenta de síntese simplificasse os circuitos.

A rotina para obter cada resultado foi colocada no item 5 “Resultados e Análises”.

Tabela 4.1– Áreas de restrição do circuito 1 no FPGA XC3S200.

ÁREA	CIRCUITO 1 CIRCUITO 2 OSCILADOR EM ANEL		CIRCUITO 1 MODIFICADO	
	CANTO INFERIOR ESQUERDO	CANTO SUPERIOR DIREITO	CANTO INFERIOR ESQUERDO	CANTO SUPERIOR DIREITO
1	X0Y0	X26Y8	X13Y4	X14Y5
2	X0Y8	X26Y16	X13Y12	X14Y13
3	X0Y16	X26Y24	X13Y20	X14Y21
4	X0Y23	X26Y31	X13Y27	X14Y28
5	X0Y31	X26Y39	X13Y35	X14Y36
6	X0Y39	X26Y47	X13Y43	X14Y44
7	X13Y0	X39Y8	X26Y4	X27Y5
8	X13Y8	X39Y16	X26Y12	X27Y13
9	X13Y16	X39Y24	X26Y20	X27Y21
10	X13Y23	X39Y31	X26Y27	X27Y28
11	X13Y31	X39Y39	X26Y35	X27Y36
12	X13Y39	X39Y47	X26Y43	X27Y44
13	X7Y0	X33Y8	X20Y4	X21Y5
14	X7Y8	X33Y16	X20Y12	X21Y13
15	X7Y16	X33Y24	X20Y20	X21Y21
16	X7Y23	X33Y31	X20Y27	X21Y28
17	X7Y31	X33Y39	X20Y35	X21Y36
18	X7Y39	X33Y47	X20Y43	X21Y44
19	X0Y0	X8Y26	X4Y13	X5Y14
20	X0Y21	X8Y47	X4Y34	X5Y35
21	X8Y0	X16Y26	X12Y13	X13Y14
22	X8Y21	X16Y47	X12Y34	X13Y35
23	X16Y0	X24Y26	X20Y13	X21Y14
24	X16Y21	X24Y47	X20Y34	X21Y35
25	X23Y0	X31Y26	X27Y13	X28Y14
26	X23Y21	X31Y47	X27Y34	X28Y35
27	X31Y0	X39Y26	X35Y13	X36Y14
28	X31Y21	X39Y47	X35Y34	X36Y35
29	X0Y11	X8Y37	X4Y24	X5Y25
30	X8Y11	X16Y37	X12Y24	X13Y25
31	X16Y11	X24Y37	X20Y24	X21Y25
32	X23Y11	X31Y37	X27Y24	X28Y25
33	X31Y11	X39Y37	X35Y24	X36Y25

Tabela 4.2 – Áreas de restrição do circuito 1 no FPGA EP2C35F672.

ÁREA	CIRCUITO 1 CIRCUITO 2 OSCILADOR EM ANEL			CIRCUITO 1 MODIFICADO		
	CANTO INFERIOR ESQUERDO	LARGURA	ALTURA	CANTO INFERIOR ESQUERDO	LARGURA	ALTURA
1	X1Y1	37	7	X19Y4	1	1
2	X1Y8	37	7	X19Y11	1	1
3	X1Y15	37	7	X19Y18	1	1
4	X1Y22	37	7	X19Y25	1	1
5	X1Y29	37	7	X19Y32	1	1
6	X28Y1	37	7	X46Y4	1	1
7	X28Y8	37	7	X46Y11	1	1
8	X28Y15	37	7	X46Y18	1	1
9	X28Y22	37	7	X46Y25	1	1
10	X28Y29	37	7	X46Y32	1	1
11	X14Y1	37	7	X32Y4	1	1
12	X14Y8	37	7	X32Y11	1	1
13	X14Y15	37	7	X32Y18	1	1
14	X14Y22	37	7	X32Y25	1	1
15	X14Y29	37	7	X32Y32	1	1
16	X1Y1	15	19	X8Y10	1	1
17	X1Y17	15	19	X8Y26	1	1
18	X13Y1	15	19	X20Y10	1	1
19	X13Y17	15	19	X20Y26	1	1
20	X26Y1	15	19	X33Y10	1	1
21	X26Y17	15	19	X33Y26	1	1
22	X38Y1	15	19	X45Y10	1	1
23	X38Y17	15	19	X45Y26	1	1
24	X50Y1	15	19	X57Y10	1	1
25	X50Y17	15	19	X57Y26	1	1
26	X1Y9	15	19	X8Y18	1	1
27	X13Y9	15	19	X20Y18	1	1
28	X26Y9	15	19	X33Y18	1	1
29	X38Y9	15	19	X45Y18	1	1
30	X50Y9	15	19	X57Y18	1	1

### 4.3 Metodologia da Análise

Para analisar os dados de cada circuito foram utilizadas várias etapas, as quais são descritas nesta seção.

Os dados, inicialmente, foram apresentados na forma de um histograma com o objetivo de mostrar a distribuição de frequência dos dados obtidos. Em seguida, foram

calculados a média e o desvio padrão dos dados obtidos nos ensaios. A partir desses valores foi estudada a dispersão dos resultados a nível de circuito e de porta lógica.

Após, os dados foram exibidos na forma de um gráfico 3D para avaliar a variação da correlação espacial local e a variação aleatória local no FPGA.

Na seqüência, aplicaram-se os métodos da média móvel de ordem 25 (NEUBERGER, 2006; SPIEGEL, 1994) e do plano de mínimos quadrados (SPIEGEL, 1994) para fazer uma transformação matemática nos dados obtidos para avaliar a correlação espacial local e a variabilidade randômica (NEUBERGER, 2006; SPIEGEL, 1994). No gráfico 3D da média móvel, o valor de cada bloco lógico programável (CLB - *Configurable Logic Block*) ou bloco de matriz lógica (LAB - *Logic Array Block*) medido é substituído pelo valor médio dos 25 valores medidos localizados ao seu redor, incluindo o valor medido do próprio CLB ou LAB que será modificado. Em outras palavras, cada CLB ou LAB receberá o valor médio de uma matriz 5x5 cujo centro é o próprio receptor original. Para os dados das bordas da média móvel foram utilizados um número menor de valores para o cálculo da média. Também verifica-se a adequação dos modelos adotados através da construção do gráfico 3D dos resíduos.

Por fim, foi utilizado o teste estatístico de hipótese para médias t-student (SPIEGEL, 1994) do programa GraphPad Prism para comparar os resultados do circuito 2 e do oscilador em anel. Nessa etapa também foram construídos gráficos em duas dimensões para facilitar a compreensão dos dados acima.

## 5 RESULTADOS E ANÁLISES

### 5.1 Circuito 1 Implementado no FPGA 2XC3S200

Para essa implementação foi utilizado o diagrama esquemático da fig. 5.1 com dois caminhos sobre teste e 110 inversores em série para cada caminho sobre teste. O apêndice A.1 contém a descrição do circuito em VHDL com a correspondente associação das chaves e leds utilizados.

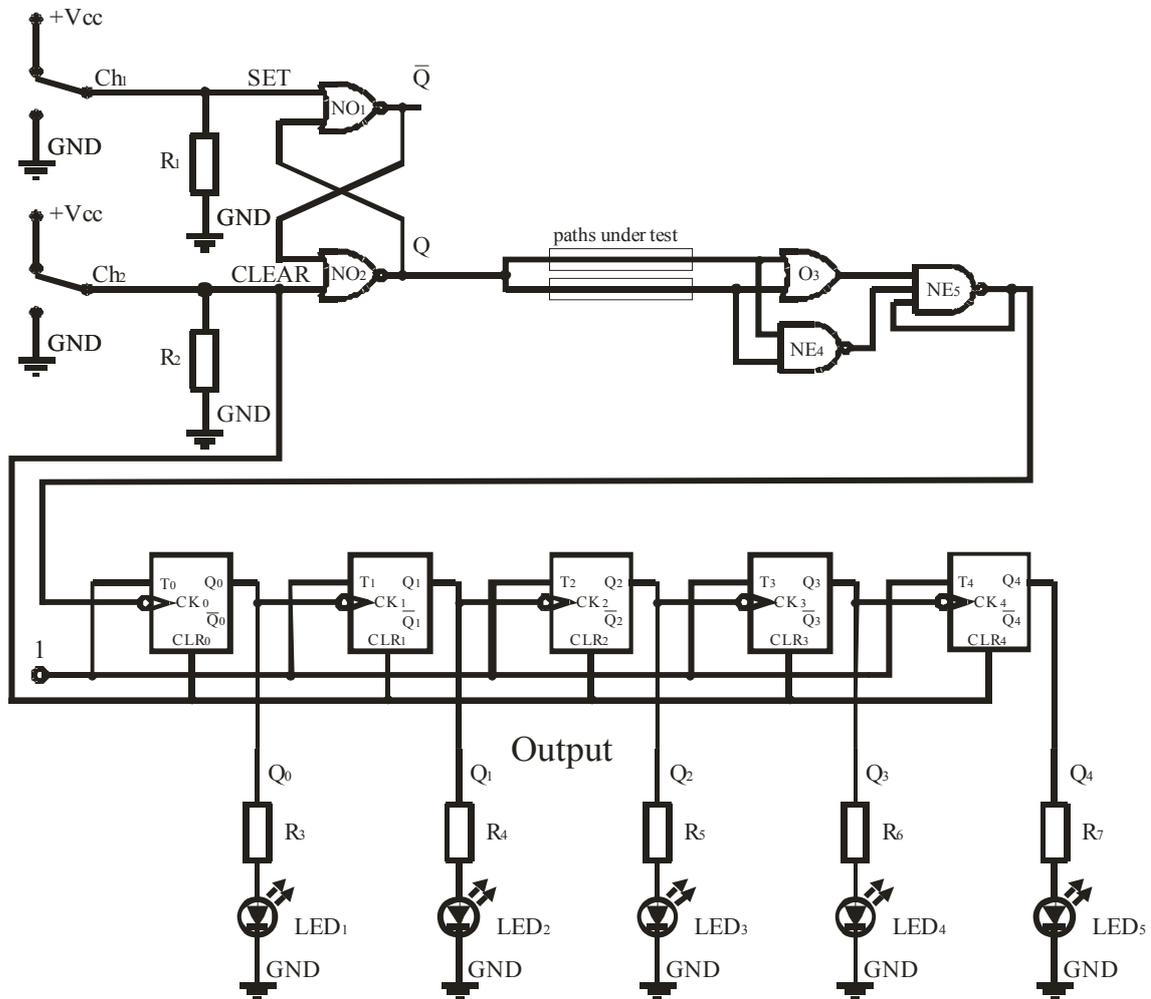
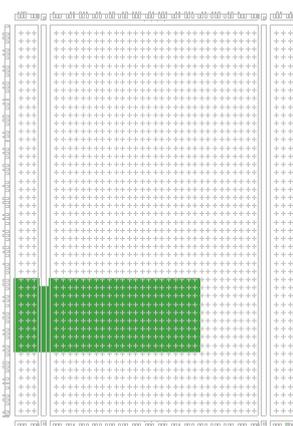
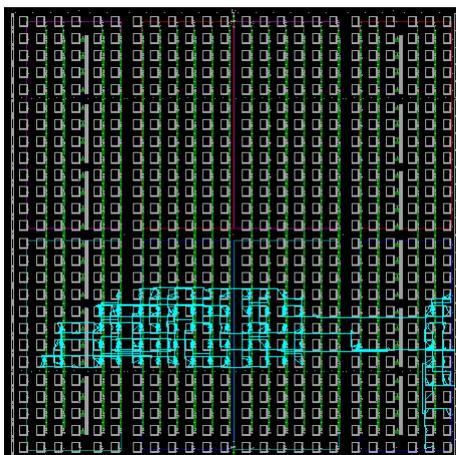


Figura 5.1 – Diagrama esquemático do circuito 1 prototipado na placa Spartan-3.

A figura 5.2 mostra a vista obtida com o programa Floorplanner após o Place & Route da área utilizada no FPGA pelo circuito 1 colocado na posição 2 e a figura 5.3 apresenta a vista obtida com o programa FPGA Editor após o Place & Route do circuito 1 colocado na posição 2 do FPGA.



**Figura 5.2 – Vista da área utilizada no FPGA pelo circuito 1 colocado na posição 2.**



**Figura 5.3 – Vista do circuito 1 colocado na posição 2 do FPGA.**

A tabela 5.1 apresenta o número de pulsos medidos pelo contador da figura 5.1 para os dois caminhos sobre teste colocados nas 33 posições indicadas pela segunda e terceira coluna da tabela 4.1. Durante essas medições a temperatura ambiente variou de 28,5°C a 29,5°C.

A rotina para obter cada resultado da tabela 5.1 iniciou com a colocação das chaves  $ch_1$  e  $ch_2$  no nível 0. Na sequência, a chave  $ch_2$  foi levada ao nível 1 e trazida ao nível 0. Após, a chave  $ch_1$  foi colocada no nível 1. Com esses procedimentos completados, é só obter o número de ciclos mostrado pelo circuito contador.

**Tabela 5.1– Resultados do circuito 1 no FPGA 2XC3S200.**

Área	Número de ciclos do sinal de saída da porta NE <sub>5</sub>
1	6
2	5
3	6
4	6
5	5
6	7
7	5
8	2
9	5
10	4
11	6
12	6
13	5
14	6
15	6
16	5
17	6
18	3
19	3
20	6
21	7
22	1
23	7
24	1
25	4
26	1
27	5
28	1
29	2
30	1
31	0
32	2
33	1

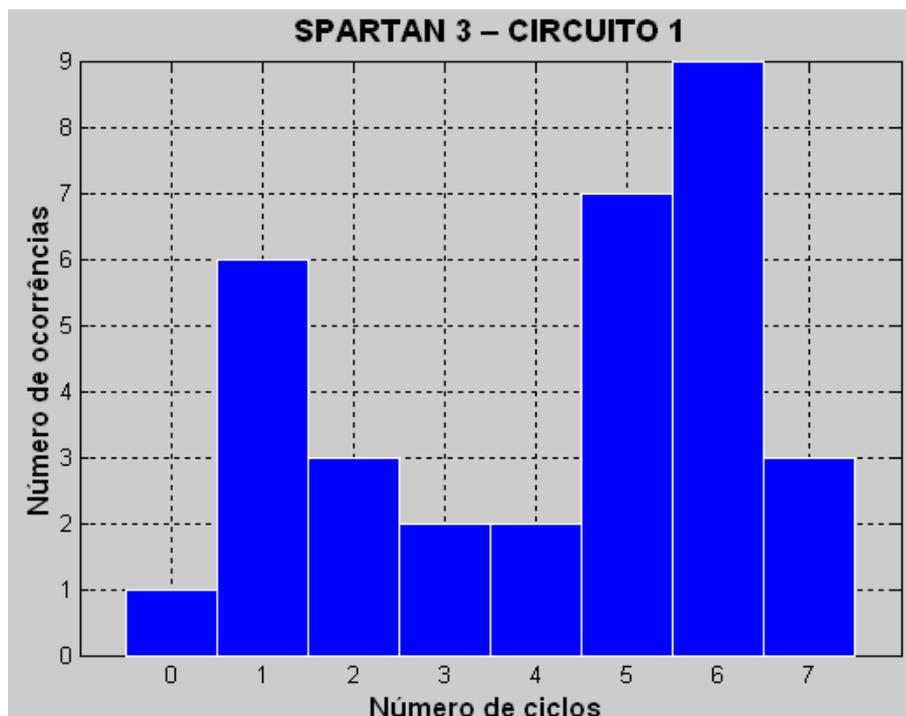
A figura 5.4 apresenta o número de ocorrências dos valores de cada área em relação ao número de ciclos do sinal de saída do circuito 1 no FPGA 2XC3S200. Usando os dados de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} = 4,12 \cong 4 \text{ ciclos}$$

Usando os valores de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}} = 2,19 = 2 \text{ ciclos}$$

Comparando o valor da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $4,12 \text{ ciclos} \pm 1,88\sigma$ .

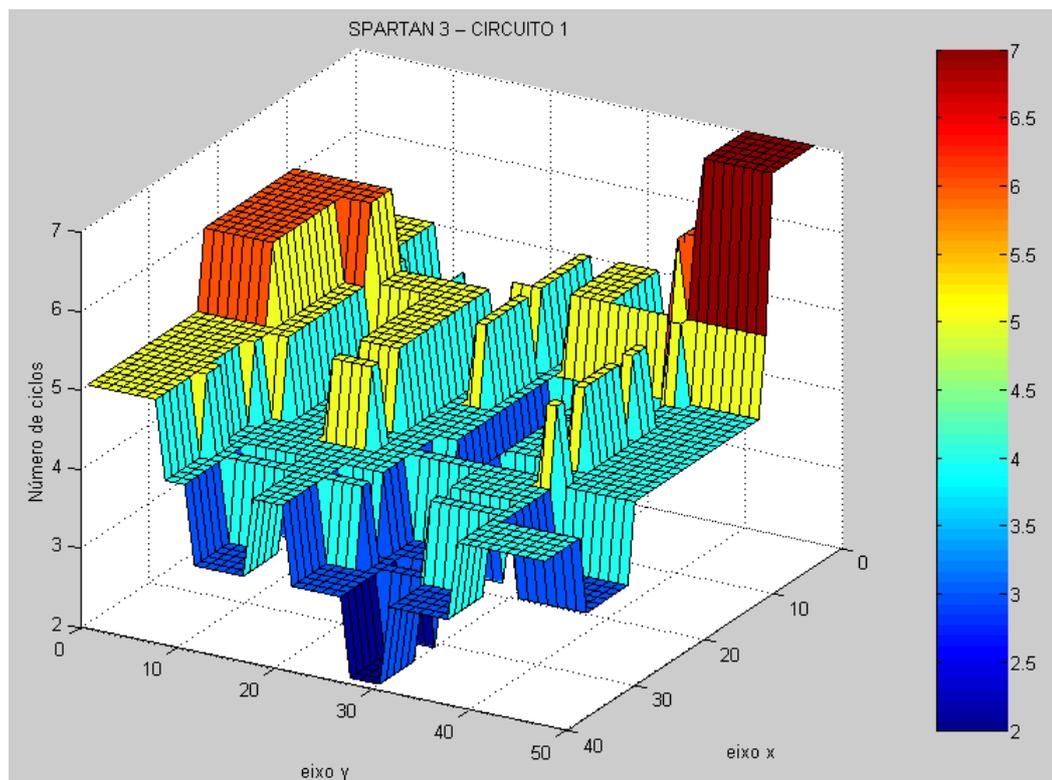


**Figura 5.4 – Número de ocorrências dos valores de cada área em relação ao número de ciclos do sinal de saída do circuito 1 no FPGA 2XC3S200.**

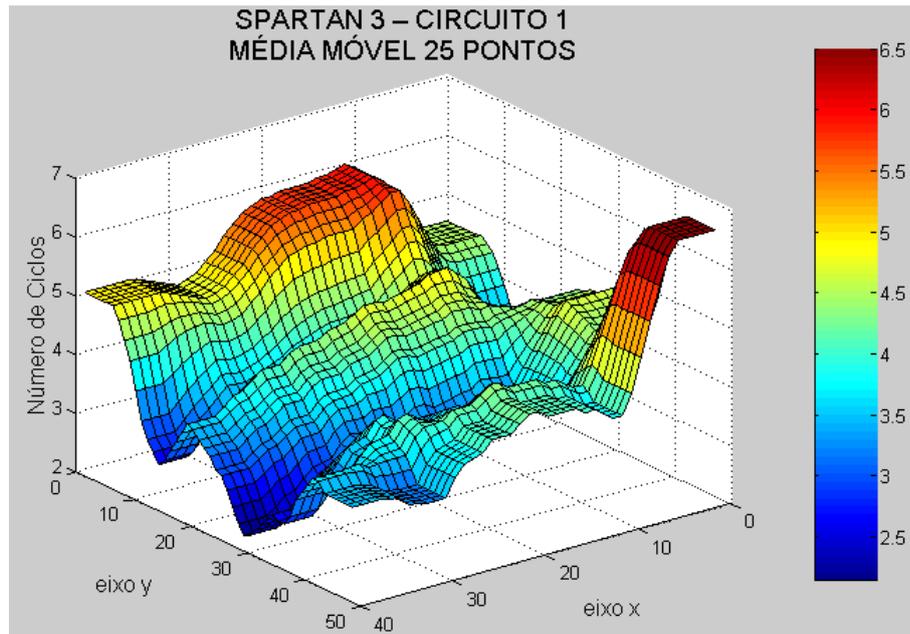
A figura 5.5 mostra a média dos valores obtidos em cada CLB do circuito 1. As figuras C.1 e C.2 do apêndice C são repetições da figura 5.5 com o observador posicionado em outro local. Analisando essa figura, nota-se que existe uma variação espacial e uma variação local aleatória dos dados obtidos. Como o circuito 1 mede a diferença entre os tempos de atraso dos caminhos sobre teste, conclui-se que as áreas com número menor de ciclos têm uma diferença menor entre os atraso de propagação das áreas medidas e que as

áreas com número maior de ciclos apresentam uma diferença maior entre os atrasos de propagação das áreas medidas. Nessa figura, a diferença entre os atrasos de propagação é predominantemente menor nos CLBs onde a coordenada x varia de 9 a 39 e a coordenada y varia de 27 a 30 e em algumas áreas adjacentes. A diferença entre os atrasos de propagação é predominantemente maior nos CLBs onde a coordenada y é 0. A diferença entre os atrasos de propagação é máxima nos CLBs onde a coordenada x varia de 0 a 6 e a coordenada y varia de 40 a 47 e é mínima nos CLBs onde a coordenada x varia de 34 a 39 e a coordenada y varia de 27 a 30.

A figura 5.6 exibe o resultado da aplicação da média móvel de ordem 25 sobre a média dos valores obtidos em cada CLB do circuito 1. Nessa figura, observa-se que o método atenuou as flutuações da figura 5.5 e salientou as tendências de variação espacial do circuito.



**Figura 5.5 – Número de ciclos do sinal de saída do circuito 1 associado às coordenadas do FPGA 2XC3S200.**



**Figura 5.6 – Média móvel dos dados do circuito 1 do FPGA 2XC3S200.**

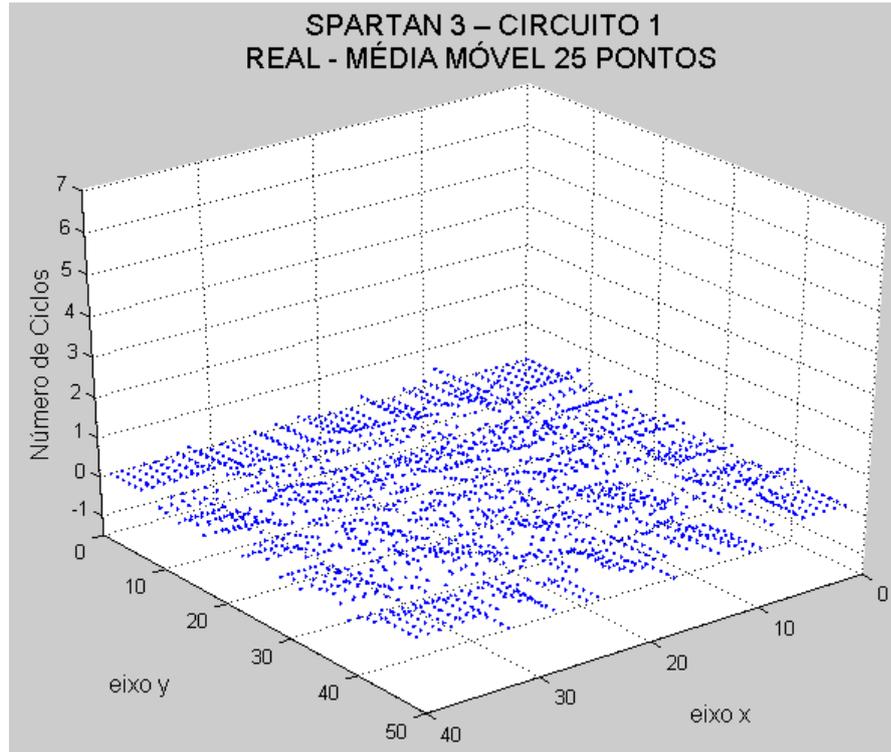
A figura 5.7 apresenta o resíduo da comparação das figuras 5.5 e 5.6. Ou seja, a diferença entre a média dos valores obtidos em cada CLB (figura 5.5) do circuito 1 e o resultado da aplicação da média móvel de ordem 25 sobre a média dos valores obtidos em cada CLB (figura 5.6) do circuito 1. Nessa figura, o maior valor de resíduo equivale a 15,57% do maior valor medido. Portanto, nesse circuito o método da média móvel de ordem 25 não mostra uma clara correlação espacial no comportamento do circuito. Isto indica a existência de uma grande componente randômica na variabilidade.

Usando a média dos valores obtidos em cada CLB do circuito 1, o sistema de equações E.4 e a equação E.3 do apêndice E, tem-se que o plano de mínimos quadrados que se ajusta ao conjunto de valores é:

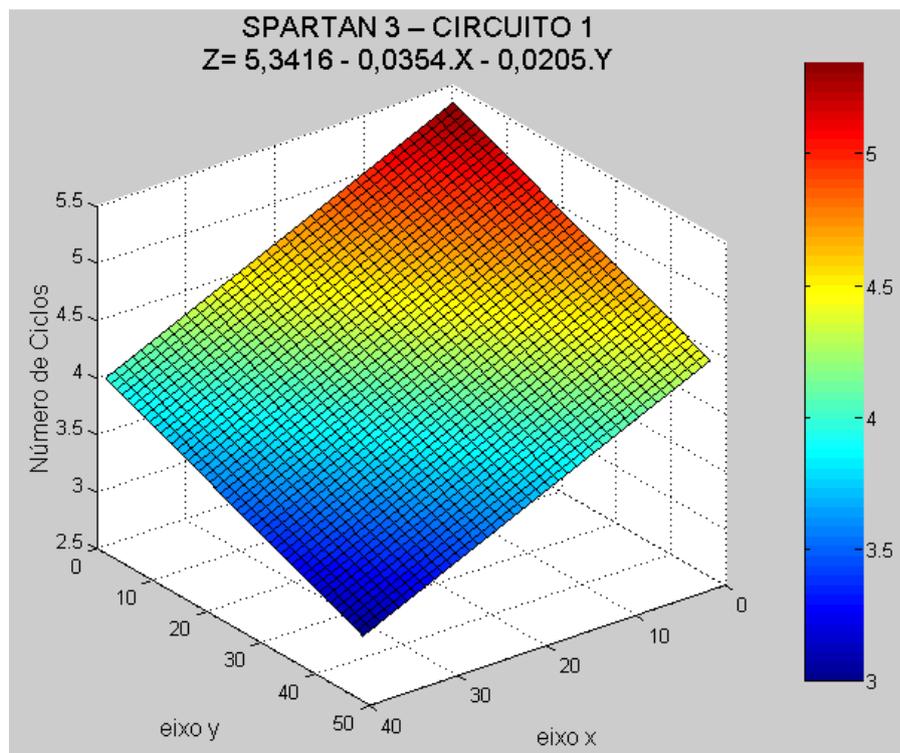
$$Z = 5,3416 - 0,0354.X - 0,0205.Y \quad (5.1)$$

Nesta equação, o valor de Z representa o valor do número de ciclos do sinal de saída obtido em cada CLB e os valores de X e Y são as coordenadas de cada CLB.

O plano de mínimos quadrados da equação 5.1, desenhado na figura 5.8, mostra a tendência da correlação espacial do circuito.



**Figura 5.7 – Diferença entre o conjunto de dados e a média móvel do circuito 1 do FPGA 2XC3S200.**

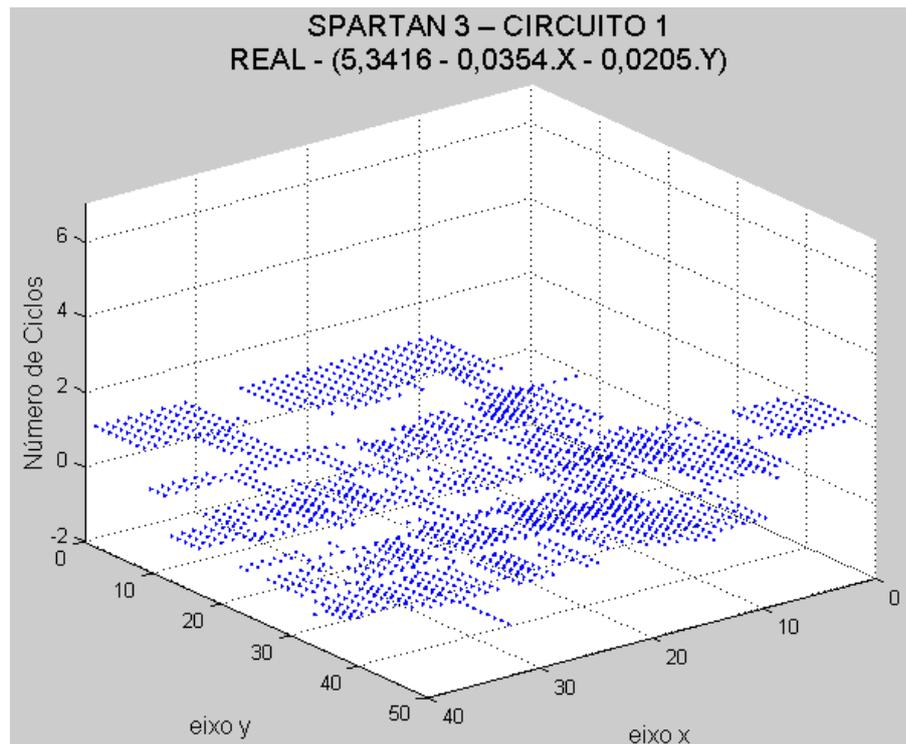


**Figura 5.8 – Plano de mínimos quadrados dos dados do circuito 1 do FPGA 2XC3S200.**

A figura 5.9 exibe o resíduo resultante das figuras 5.5 e 5.8. Nessa figura, observa-se que o maior valor do resíduo equivale a 28,57% do maior valor medido. Portanto, nesse

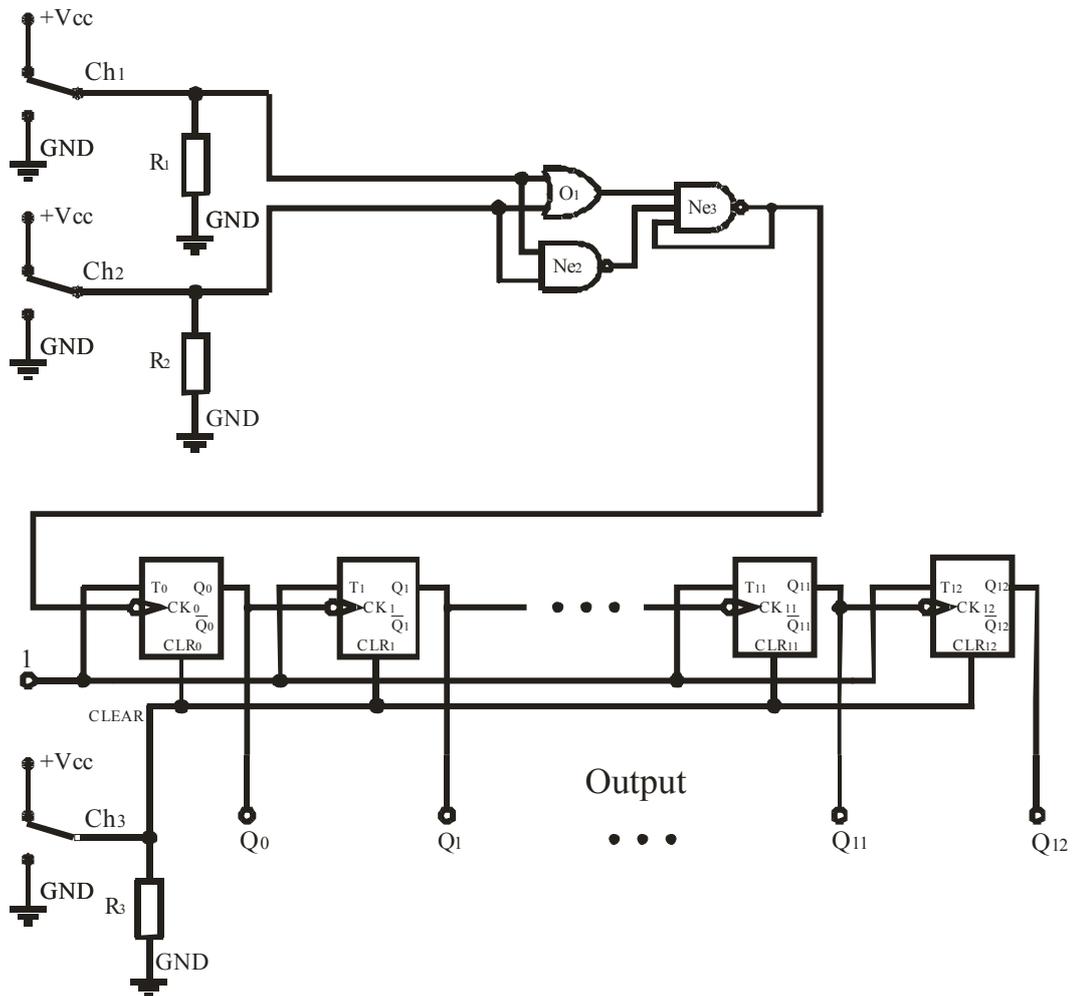
circuito o gráfico do plano de mínimos quadrados também indica a existência de uma grande componente randômica na variabilidade desse circuito.

Analisando as figuras 5.5, 5.6 e 5.8, vê-se que todas mostram a mesma tendência na correlação espacial do circuito e indicam a existência de uma forte componente randômica na variabilidade desse circuito.



**Figura 5.9 – Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 1 do FPGA 2XC3S200.**

Para obter a frequência de oscilação da porta  $Ne_5$  do circuito 1 e o tempo de atraso de um flip-flop, o diagrama esquemático do circuito 1 foi modificado (figura 5.10). A descrição em VHDL desse circuito está no apêndice A.3. A figura 5.11 apresenta as formas de onda das saídas  $Q_{11}$  e  $Q_{12}$  do circuito 1 modificado implementado na segunda posição do FPGA da placa Spartan 3. E a figura 5.12 mostra as formas de onda utilizadas para medir o tempo de atraso do flip-flop 12 do circuito 1 modificado implementado na segunda posição do FPGA da placa Spartan 3.



**Figura 5.10 – Diagrama esquemático do circuito 1 modificado prototipado na placa Spartan-3.**

A tabela 5.2 mostra os tempos de atraso do flip-flop 12 ( $t_{pQ12}$ ) e os períodos das tensões das saídas  $Q_{11}$  ( $T_{Q11}$ ) e  $Q_{12}$  ( $T_{Q12}$ ) do circuito 1 modificado colocado nas 33 posições indicadas pela quarta e quinta coluna da tabela 4.1. Durante as medições, a temperatura ambiente variou de 26,0°C a 27,0°C.

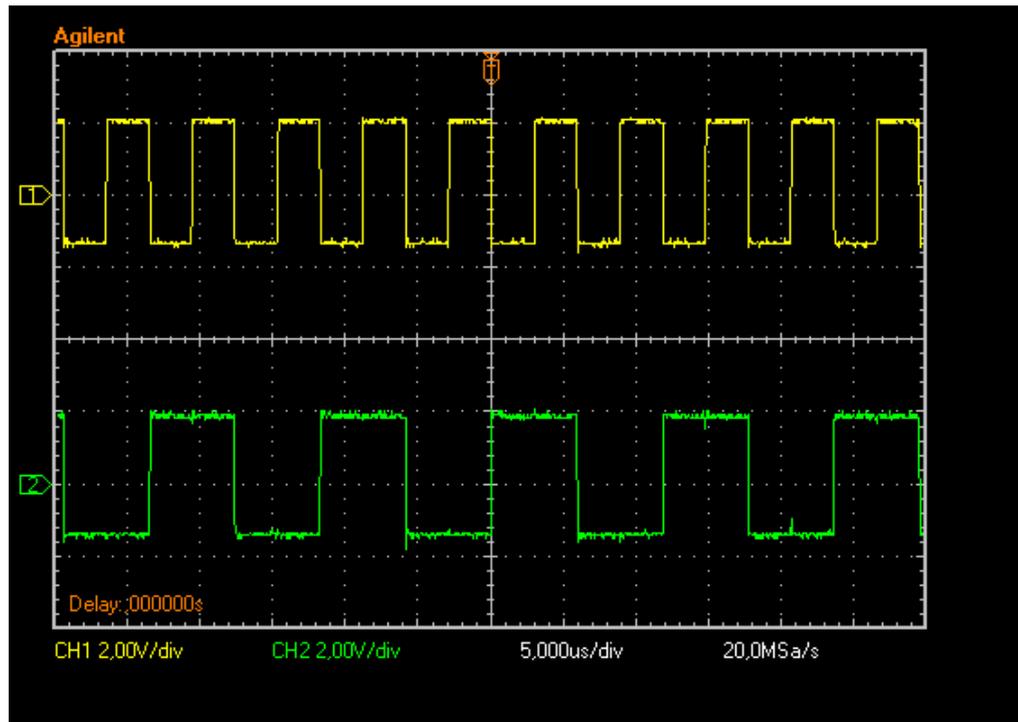


Figura 5.11 – Formas de onda das saídas  $Q_{11}$ (acima) e  $Q_{12}$  (abaixo) do circuito 1 modificado na segunda posição do FPGA XC3S200.

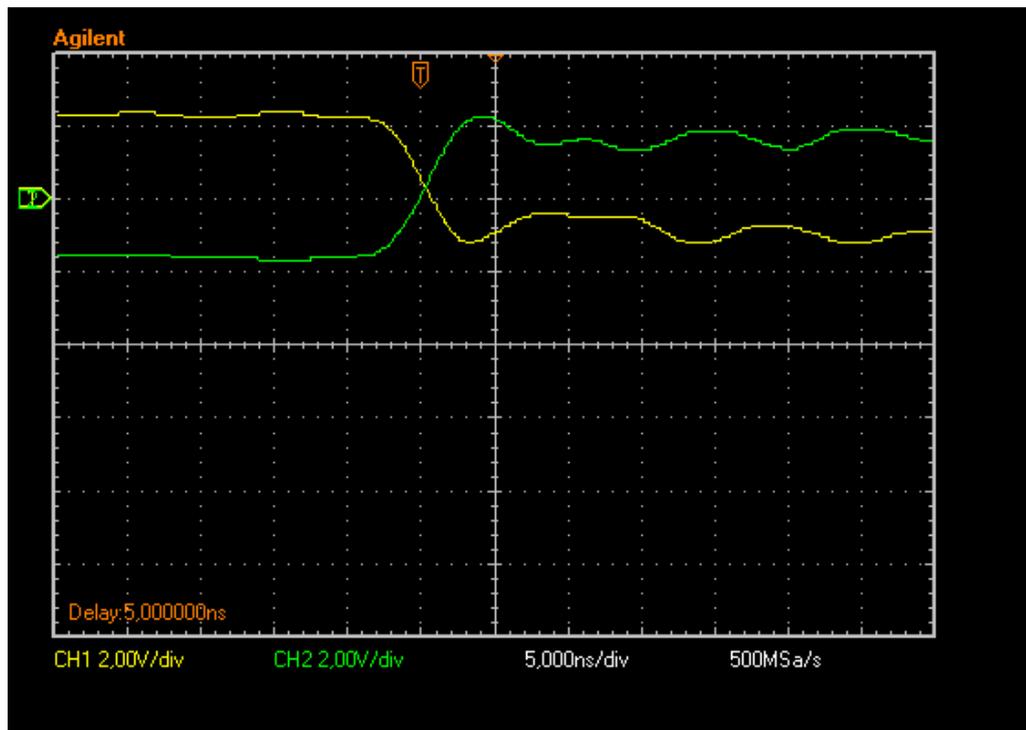


Figura 5.12 – Formas de onda utilizadas para medir o tempo de atraso do flip-flop 12 do circuito 1 modificado implementado na segunda posição do FPGA XC3S200.

Tabela 5.2 – Resultados do circuito 1 modificado no FPGA XC3S200.

Áreas	Elementos das amostras		
	$t_{pQ12}$ (ns)	$T_{Q11}$ ( $\mu$ s)	$T_{Q12}$ ( $\mu$ s)
1	0,2	5,900	11,80
2	0,8	5,900	11,80
3	0,2	6,100	12,20
4	0,2	7,200	14,40
5	0,2	7,160	14,32
6	0,2	7,400	14,80
7	0,8	6,140	12,28
8	0,5	6,200	12,35
9	0,5	5,880	11,76
10	0,5	7,060	14,12
11	0,8	7,040	14,08
12	0,8	7,000	14,00
13	0,5	6,240	12,48
14	0,2	6,060	12,12
15	0,2	5,960	11,92
16	0,2	7,000	14,00
17	0,2	7,160	14,32
18	0,5	7,200	14,40
19	0,8	5,880	11,76
20	0,8	6,040	12,08
21	0,2	5,820	11,64
22	1,0	6,020	12,04
23	0,2	6,020	12,04
24	0,5	6,120	12,24
25	0,8	7,340	14,68
26	1,0	5,920	11,84
27	0,7	6,580	13,16
28	0,8	6,220	12,44
29	0,2	6,340	12,68
30	0,2	6,380	12,76
31	0,1	6,420	12,84
32	0,3	6,320	12,64
33	0,1	6,420	12,84

Para obter cada resultado da tabela 5.2 é necessário colocar a chave  $ch_1$  no nível 1, a chave  $ch_2$  no nível 0, a chave  $ch_3$  no nível 0 e medir o tempo de atraso do flip-flop 12 e o período dos sinais das saídas  $Q_{11}$  e  $Q_{12}$  com o osciloscópio.

A partir dos dados de  $Q_{12}$  da tabela 5.2 e da fórmula E.1 do apêndice E, tem-se que o valor médio do período de todo o conjunto de dados de  $Q_{12}$  ( $\bar{T}_{Q_{12}}$ ) é:

$$\bar{T}_{Q_{12}} = \frac{\sum_{i=1}^n T_i}{n} = 12,87 \mu\text{s}$$

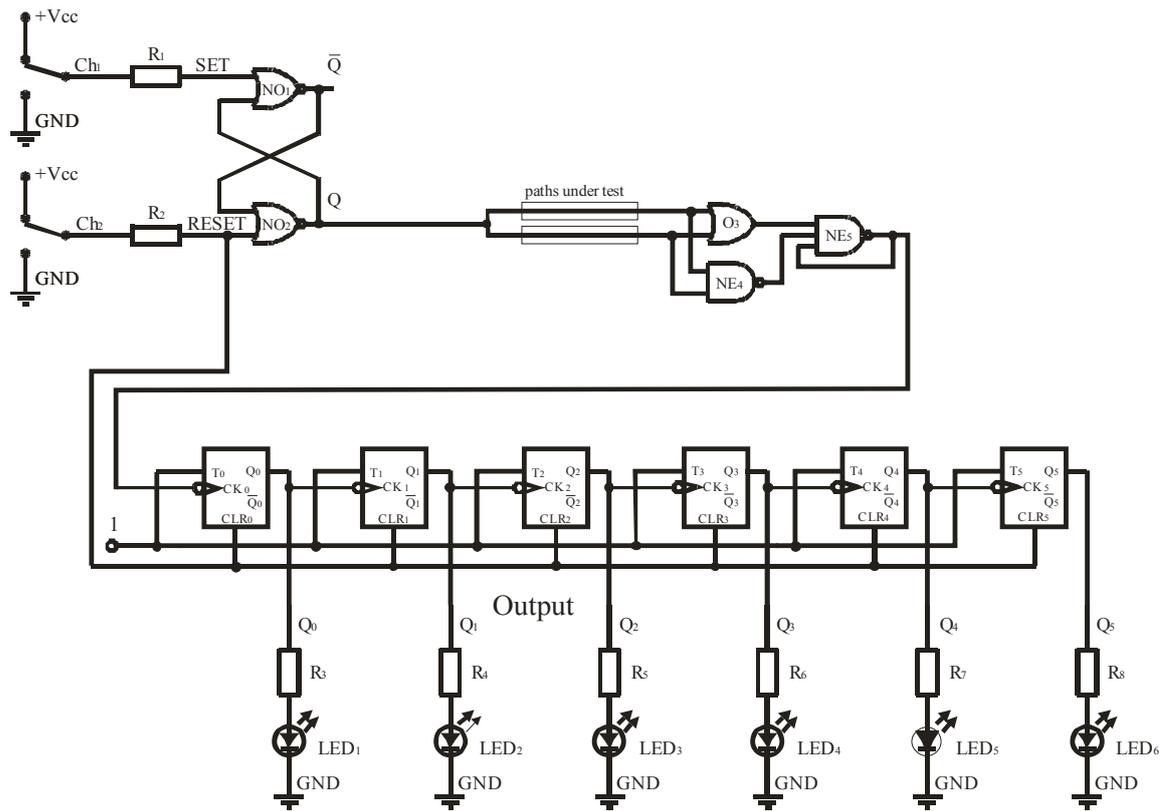
Com o valor médio do período de  $Q_{12}$ , calculou-se que a frequência de saída da porta  $NE_3$  é de 636,34 MHz ( $T=1,57\text{ns}$ ).

A partir dos dados do tempo de atraso do flip-flop 12 da tabela 5.2 e da fórmula E.1 do apêndice E, tem-se que o valor médio do tempo de atraso do flip-flop 12 ( $\bar{t}_{pQ_{12}}$ ) é:

$$\bar{t}_{pQ_{12}} = \frac{\sum_{i=1}^n t_{pQ_{12i}}}{n} = 0,46 \text{ ns}$$

## 5.2 Circuito 1 Implementado no FPGA EP2C35F672

Para essa implementação foi utilizado o diagrama esquemático da figura 5.13 com dois caminhos sobre teste e 1350 inversores em série para cada caminho sobre teste. Esse número de inversores é devido ao fato de o EP2C35F672 ter maior número de LAB por unidade de área. O apêndice A.2 contém a descrição do circuito em VHDL com a correspondente associação das chaves e leds utilizados.



**Figura 5.13 – Diagrama esquemático do circuito 1 prototipado na placa Cyclone II.**

Encontra-se, na figura 5.14, a vista obtida com o programa Chip Planner da área utilizada no FPGA pelo circuito 1 colocado na posição 16.



**Figura 5.14 – Vista da área utilizada no FPGA pelo circuito 1 colocado na posição 16.**

A tabela 5.3 mostra o número de pulsos medidos pelo contador do circuito da figura 5.13 para os dois caminhos sobre teste colocados nas 30 posições indicadas pela segunda, terceira e quarta coluna da tabela 4.2. Durante essas medições a temperatura ambiente variou de 24,5°C a 25,5°C.

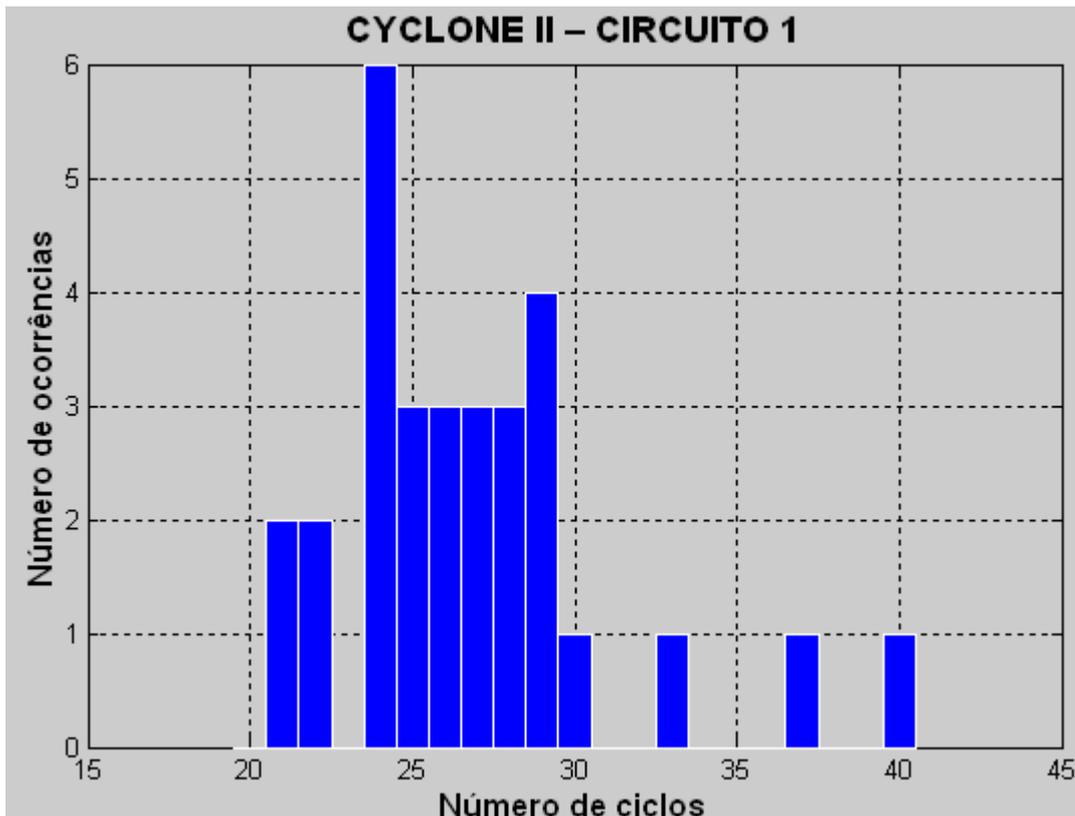
Tabela 5.3– Resultados do circuito 1 no FPGA EP2C35F672.

<b>Áreas</b>	<b>Número de ciclos do sinal de saída da porta NE<sub>5</sub></b>
<b>1</b>	29
<b>2</b>	22
<b>3</b>	25
<b>4</b>	27
<b>5</b>	26
<b>6</b>	24
<b>7</b>	27
<b>8</b>	29
<b>9</b>	26
<b>10</b>	29
<b>11</b>	24
<b>12</b>	26
<b>13</b>	37
<b>14</b>	22
<b>15</b>	25
<b>16</b>	24
<b>17</b>	24
<b>18</b>	28
<b>19</b>	24
<b>20</b>	27
<b>21</b>	42
<b>22</b>	28
<b>23</b>	28
<b>24</b>	21
<b>25</b>	33
<b>26</b>	25
<b>27</b>	29
<b>28</b>	24
<b>29</b>	21
<b>30</b>	30

A rotina para obter cada resultado da tabela 5.3 é a mesma usada no circuito 1 no FPGA 2XC3S200.

A figura 5.15 mostra o número de ocorrências dos valores de cada área em relação ao número de ciclos do sinal de saída do circuito 1. Usando os dados de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} = 27 \text{ ciclos}$$



**Figura 5.15 – Número de ocorrências dos valores de cada área em relação ao número de ciclos do sinal de saída do circuito 1 no FPGA EP2C35F672.**

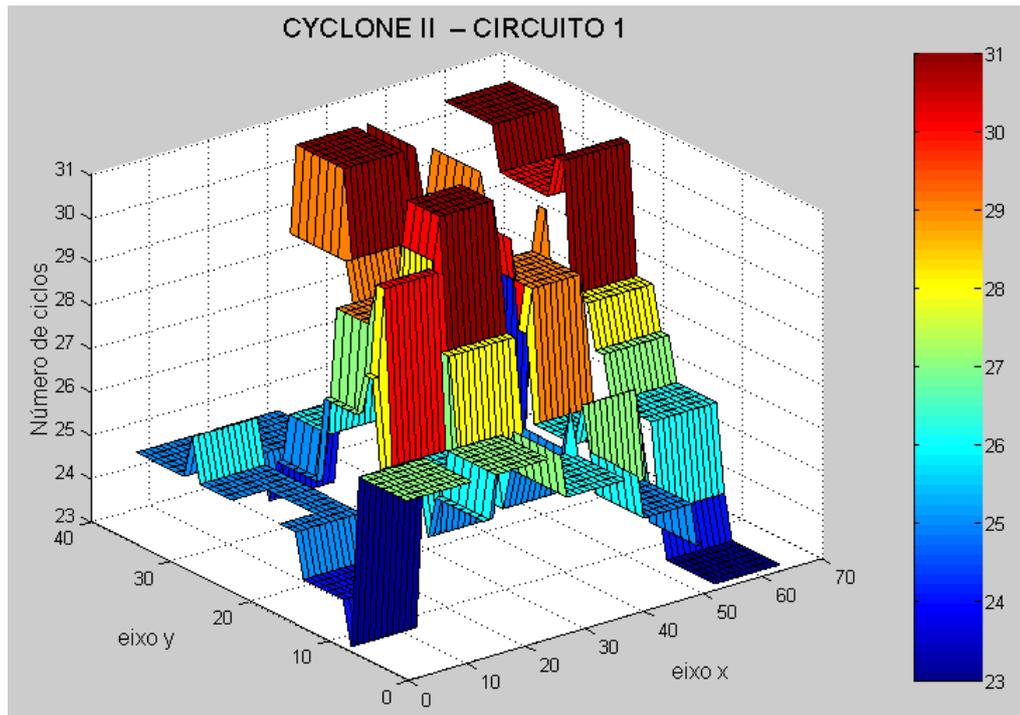
Usando os valores de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}} = 4,47 \cong 4 \text{ ciclos}$$

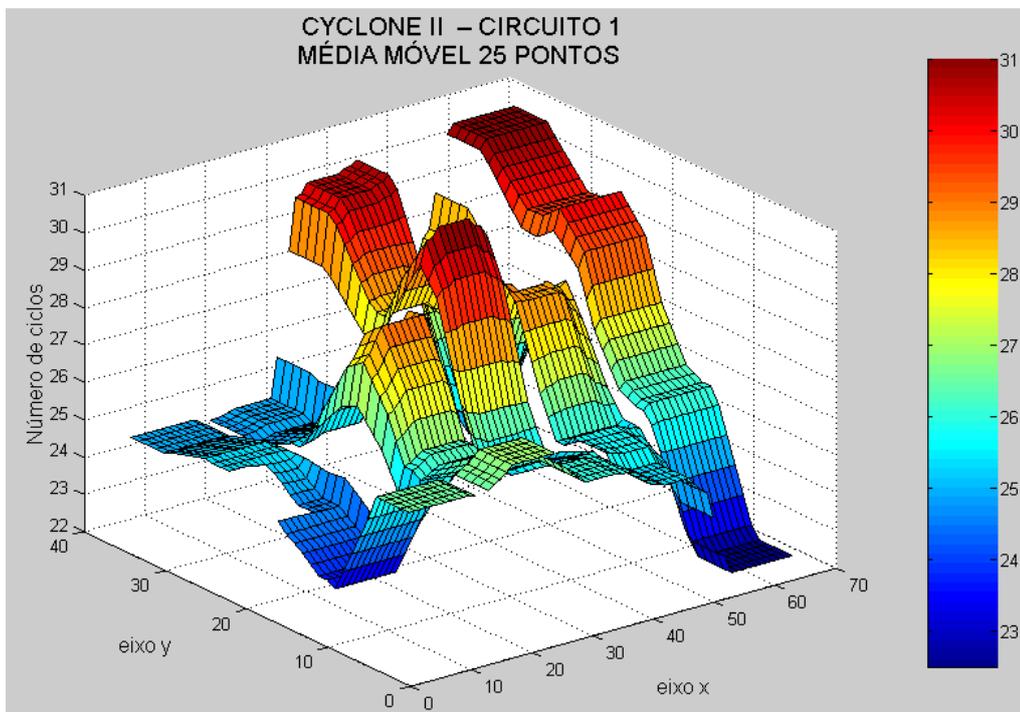
Comparando o valor da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de 27 ciclos  $\pm 3,39\sigma$ .

A figura 5.16 apresenta a média dos valores obtidos em cada LAB do circuito 1. Os “espaços em branco” dessa figura representam as áreas que não tem LAB disponíveis para uso. A figura C.7 do apêndice C é uma repetição da figura 5.16 com o observador posicionado em outro local. Analisando essa figura, observa-se que existe uma variação espacial e uma variação local aleatória dos dados obtidos. E como o circuito 1 mede a diferença entre os tempo de atraso dos caminhos sobre teste, conclui-se que as áreas com número menor de ciclos apresentam uma diferença menor entre os atraso de propagação das áreas medidas e as áreas com número maior de ciclos apresentam uma diferença maior entre os atraso de propagação das áreas medidas. Nessa figura, a diferença entre os atrasos de propagação é predominantemente menor nos LABs onde a coordenada  $x$  é 1 ou a coordenada  $y$  é 1 e que a diferença entre os atrasos de propagação é predominantemente maior nos LABs próximos ao cruzamento das coordenadas onde  $x$  é 64 e  $y$  é 35. A diferença entre os atrasos de propagação é máxima em quatro áreas: a primeira fica nos LABs onde a coordenada  $x$  varia de 53 a 64 e a coordenada  $y$  varia de 29 a 35, a segunda fica nos LABs onde a coordenada  $x$  varia de 28 a 40 e a coordenada  $y$  varia de 29 a 35, a terceira fica nos LABs onde a coordenada  $x$  varia de 53 a 64 e a coordenada  $y$  varia de 20 a 21, e a quarta fica nos LABs onde a coordenada  $x$  varia de 28 a 37 e a coordenada  $y$  varia de 17 a 21. A diferença entre os atrasos de propagação é mínima nos LABs onde a coordenada  $x$  varia de 53 a 64 e a coordenada  $y$  varia de 1 a 7.

A figura 5.17 apresenta o resultado da aplicação da média móvel de ordem 25 sobre a média dos valores obtidos em cada LAB do circuito 1. Nessa figura, visualiza-se que o método atenuou as flutuações da figura 5.16 e salientou as tendências de variação espacial do circuito.



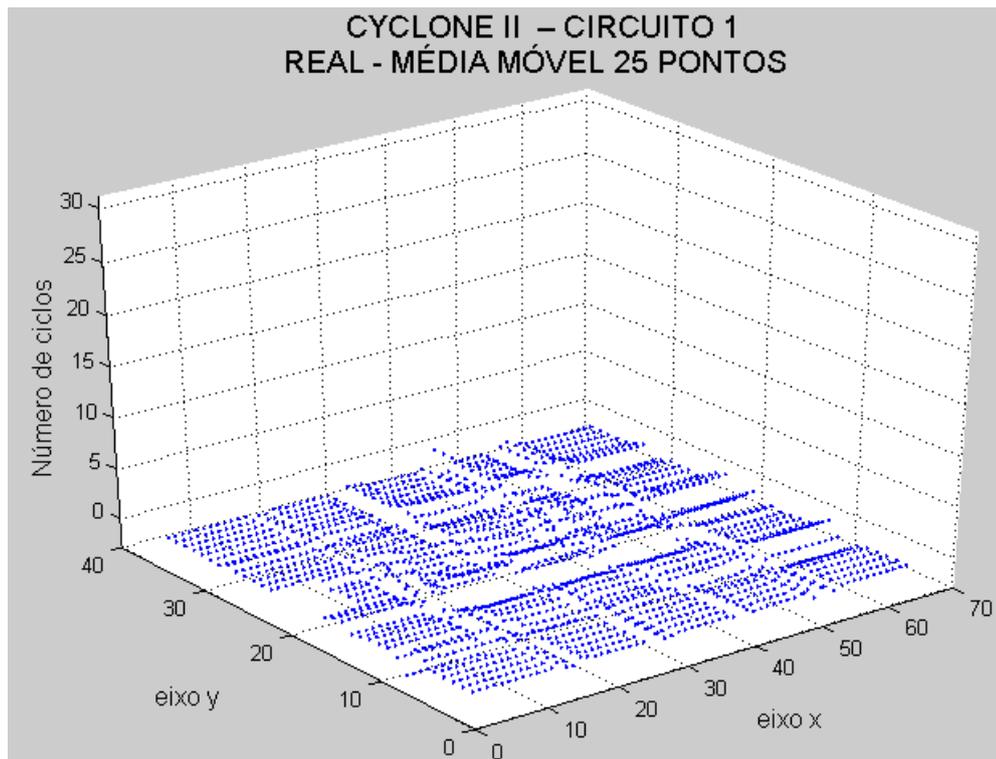
**Figura 5.16 – Número de ciclos do sinal de saída do circuito 1 associado às coordenadas do FPGA EP2C35F672.**



**Figura 5.17 – Média móvel dos dados do circuito 1 do FPGA EP2C35F672.**

A figura 5.18 mostra o resíduo da comparação das figuras 5.16 e 5.17. Ou seja, a diferença entre a média dos valores obtidos em cada LAB (figura 5.16) do circuito 1 e o

resultado da aplicação da média móvel de ordem 25 sobre a média dos valores obtidos em cada LAB (figura 5.17) do circuito 1. Nessa figura, nota-se que o maior valor de resíduo equivale a 9,3% do maior valor medido. Portanto, nesse circuito o método da média móvel de ordem 25 não exibe uma clara correlação espacial no comportamento do circuito. Isto mostra a existência de uma grande componente aleatória na variabilidade.



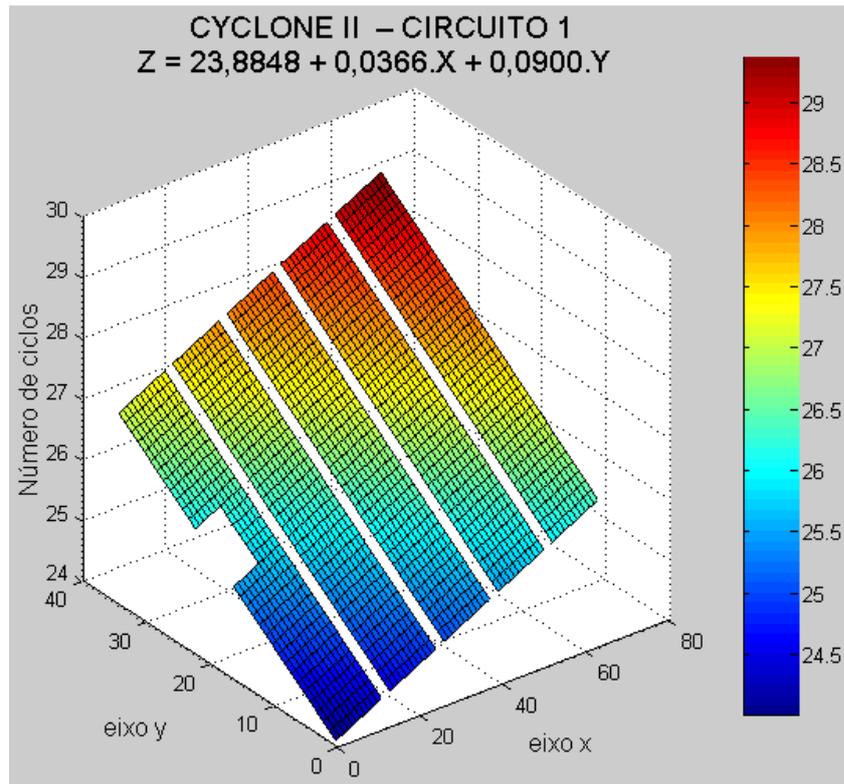
**Figura 5.18 – Diferença entre o conjunto de dados e a média móvel do circuito 1 do FPGA EP2C35F672.**

Usando a média dos valores obtidos em cada LAB do circuito 1, o sistema de equações E.4 e a equação E.3 do apêndice E, tem-se que o plano de mínimos quadrados que se ajusta ao conjunto de valores é:

$$Z = 23,8848 + 0,0366.X + 0,0900.Y \quad (5.2)$$

Nesta equação, o valor de Z representa o valor do número de ciclos do sinal de saída obtido em cada LAB e os valores de X e Y representam as coordenadas de cada LAB.

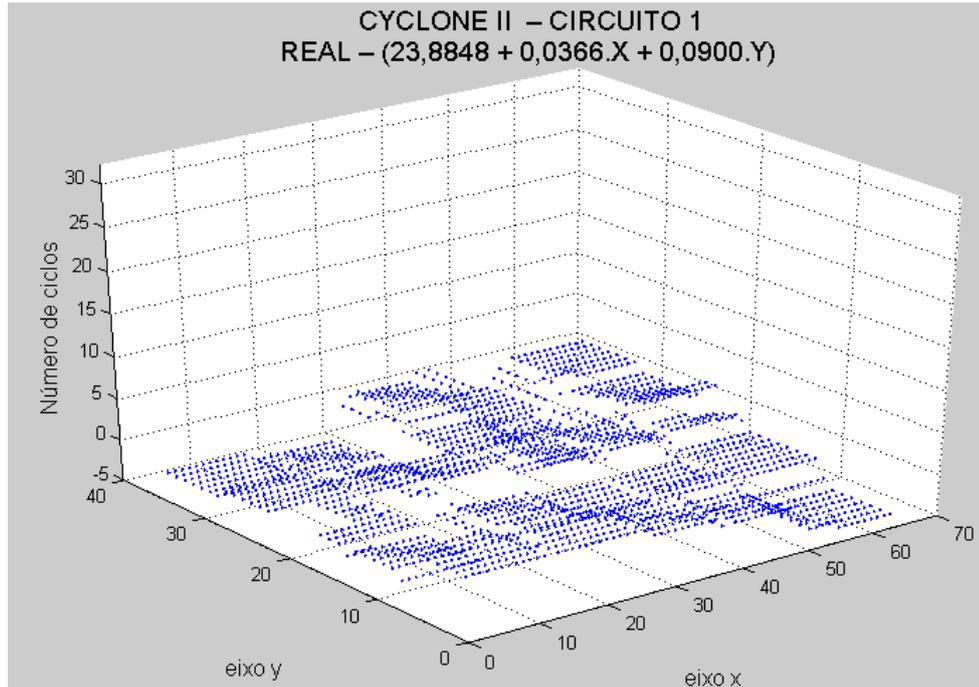
O plano de mínimos quadrados da equação 5.2, desenhado na figura 5.19, mostra a tendência da variação espacial do circuito.



**Figura 5.19 – Plano de mínimos quadrados dos dados do circuito 1 do FPGA EP2C35F672.**

A figura 5.20 apresenta o resíduo da comparação das figuras 5.16 e 5.19. Nessa figura, observa-se que o maior valor de resíduo equivale a 16,13% do maior valor medido. Portanto, nesse circuito o gráfico do plano de mínimos quadrados também indica a existência de uma grande componente randômica na variabilidade deste circuito.

Observando-se as figuras 5.16, 5.17 e 5.19, vê-se que todas mostram a mesma tendência na correlação espacial do circuito e indicam a existência de uma forte componente randômica na variabilidade desse circuito.

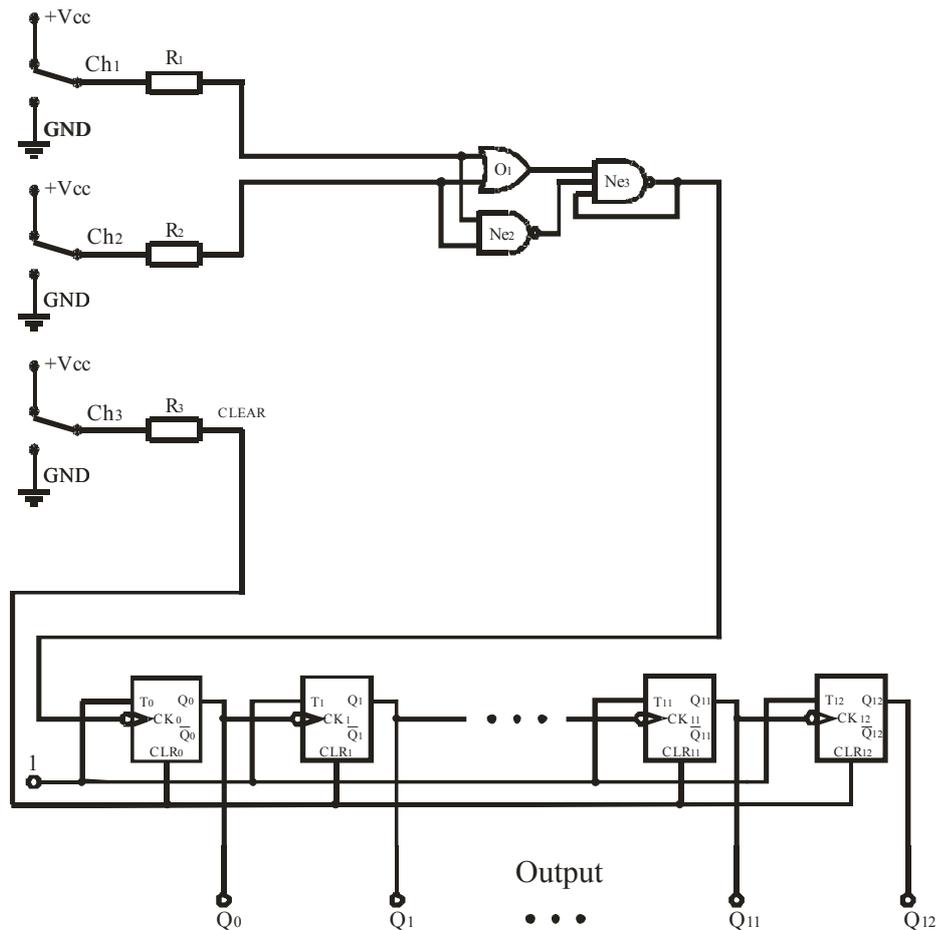


**Figura 5.20 – Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 1 do FPGA EP2C35F672.**

A figura 5.21 mostra o diagrama esquemático do circuito 1 modificado. Com essa modificação, pretende-se obter a frequência de oscilação da porta  $Ne_3$  e o tempo de atraso de um flip-flop. O apêndice A.4 contém a descrição em VHDL do circuito 1 modificado.

A tabela 5.4 apresenta os tempos de atraso do flip-flop 12 ( $t_{pQ12}$ ) e os períodos das tensões das saídas  $Q_{11}$  ( $T_{Q11}$ ) e  $Q_{12}$  ( $T_{Q12}$ ) do circuito 1 modificado colocado nas 30 posições indicadas pela quinta, sexta e sétima colunas da tabela 4.2. Durante as medições, a temperatura ambiente manteve-se em  $26,2^{\circ}\text{C}$ .

A rotina para obter cada resultado apresentado da tabela 5.4 é a mesma usada no circuito 1 modificado prototipado no FPGA 2XC3S200.



**Figura 5.21 – Diagrama esquemático do circuito 1 modificado prototipado na placa Cyclone II.**

A partir dos dados de  $Q_{12}$  da tabela 5.4 e da fórmula E.1 do apêndice E, tem-se que o valor médio do período de todo o conjunto de dados de  $Q_{12}$  ( $\bar{T}_{Q_{12}}$ ) é:

$$\bar{T}_{Q_{12}} = \frac{\sum_{i=1}^n T_i}{n} = 5,40 \mu\text{s}$$

Assim, a frequência de saída da porta NE<sub>3</sub>, calculada com o valor médio do período de  $Q_{12}$ , é de 1,51 GHz ( $T=0,66\text{ns}$ ).

A partir dos dados do tempo de atraso do flip-flop 12 da tabela 5.4 e da fórmula E.1 do apêndice E, tem-se que o valor médio do tempo de atraso do flip-flop 12 ( $\bar{t}_{pQ_{12}}$ ) é:

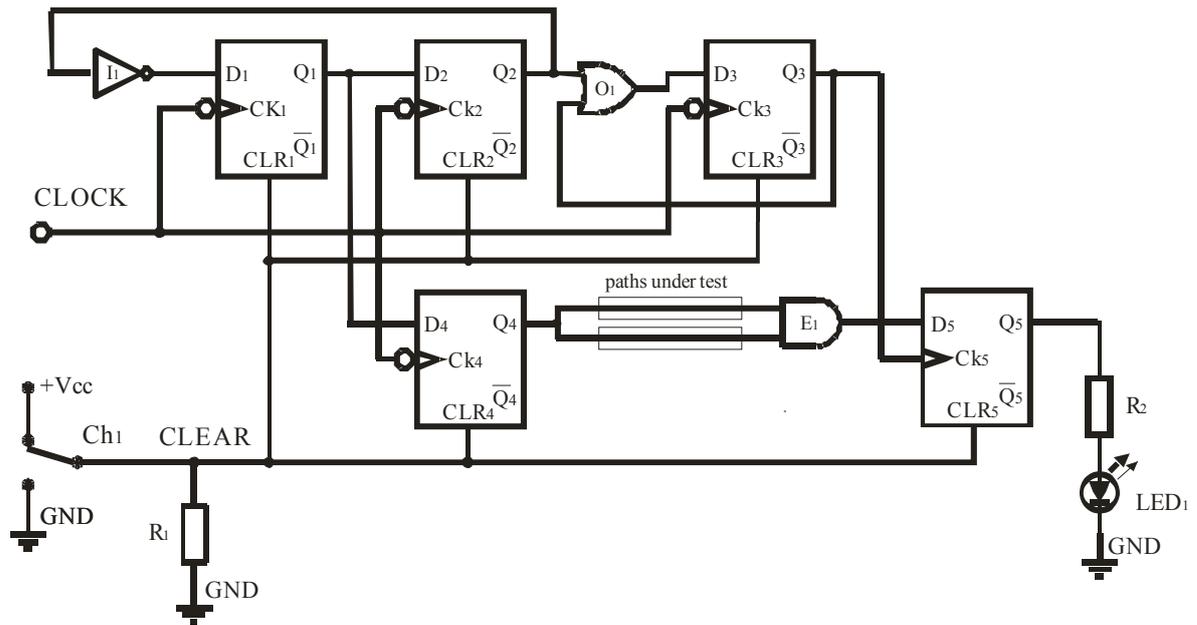
$$\bar{t}_{pQ_{12}} = \frac{\sum_{i=1}^n t_{pQ_{12i}}}{n} = 1,30 \text{ ns}$$

Tabela 5.4– Resultados do circuito 1 modificado no FPGA EP2C35F672.

Áreas	Elementos das amostras		
	$t_{pQ12}$ (ns)	$T_{Q11}$ ( $\mu$ s)	$T_{Q12}$ ( $\mu$ s)
1	1,5	2,520	5,060
2	1,0	2,520	5,040
3	1,5	2,540	5,080
4	1,0	2,500	5,000
5	1,0	2,500	5,000
6	1,5	3,560	7,120
7	1,8	2,540	5,080
8	1,3	2,540	5,080
9	1,5	2,520	5,040
10	1,0	2,520	5,040
11	1,5	2,540	5,080
12	1,0	2,520	5,040
13	1,1	2,500	5,000
14	1,0	3,260	6,520
15	1,0	3,740	7,480
16	1,2	2,540	5,080
17	0,8	3,320	6,640
18	1,7	2,520	5,040
19	1,5	2,500	5,000
20	1,2	2,500	5,000
21	1,0	2,480	4,960
22	1,9	3,300	6,600
23	1,0	2,540	5,080
24	1,8	2,540	5,080
25	1,4	2,540	5,100
26	1,2	2,520	5,040
27	1,5	2,480	4,960
28	1,2	2,520	5,040
29	1,0	3,280	6,560
30	1,8	2,540	5,080

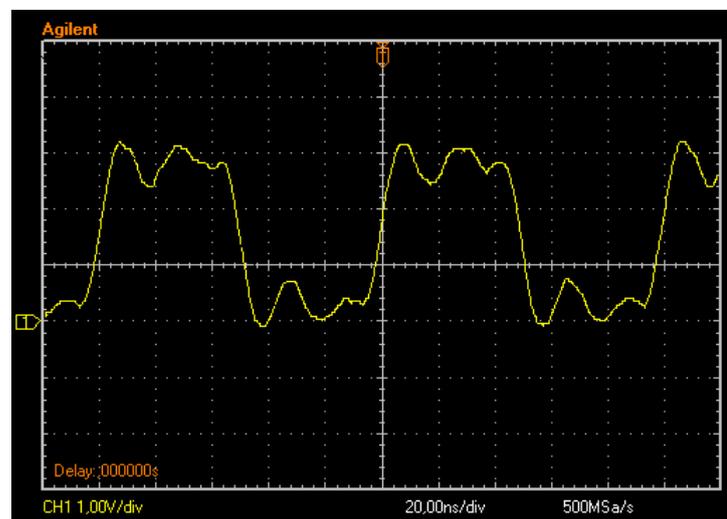
### 5.3 Circuito 2 Implementado no FPGA 2XC3S200

Para essa implementação foi utilizado o diagrama esquemático da fig. 5.22 com dois caminhos sobre teste e 110 inversores em série para cada caminho sobre teste. O apêndice A.5 contém a descrição em VHDL do circuito.



**Figura 5.22 – Diagrama esquemático do circuito 2 prototipado na placa Spartan-3.**

A figura 5.23 apresenta a forma de onda aplicada na entrada *clock* do circuito 2 implementado na placa Spartan 3. As distorções mostradas nessa figura não foram corrigidas mesmo após a ponteira de prova ter sido ajustada conforme o manual do osciloscópio.



**Figura 5.23 – Forma de onda aplicada na entrada do circuito 2 implementado na área 2 do FPGA da placa Spartan 3.**

A tabela 5.5 apresenta os resultados das medidas efetuadas no circuito da figura 5.22 para os dois caminhos sobre teste colocados nas 33 posições indicadas pela segunda e terceira coluna da tabela 4.1. Durante as medições, a temperatura ambiente variou de 24,0°C a 26,0°C.

A rotina usada para encontrar o valor da frequência do *clock* que faz o circuito mudar o estado do led<sub>1</sub>, de apagado para aceso é mostrada através de um exemplo. Inicialmente, pretende-se supor que essa frequência desconhecida seja o primeiro valor da tabela 5.5, ou seja, 10,054 MHz. Então, tem-se duas opções ou o gerador de funções tem um valor de frequência que é menor ou igual ao valor da frequência desconhecida e, portanto, acenderá o led<sub>1</sub> ou tem um valor de frequência que é maior ao valor da frequência desconhecida e, portanto, não acenderá o led<sub>1</sub>. No experimento, o primeiro ajuste do gerador de funções sempre foi para um valor que mantém o led apagado.

**Tabela 5.5– Resultados das medidas efetuadas com o circuito 2 no FPGA 2XC3S200.**

Áreas	Elementos das amostras (MHz)									
	1	2	3	4	5	6	7	8	9	10
1	10,054	10,054	10,054	10,055	10,054	10,054	10,054	10,054	10,055	10,054
2	11,863	11,868	11,867	11,868	11,865	11,865	11,865	11,865	11,864	11,865
3	12,278	12,277	12,279	12,272	12,272	12,271	12,272	12,273	12,271	12,271
4	12,230	12,231	12,230	12,231	12,231	12,229	12,229	12,229	12,230	12,229
5	12,311	12,310	12,311	12,311	12,308	12,309	12,306	12,306	12,308	12,308
6	11,570	11,570	11,565	11,568	11,569	11,574	11,571	11,570	11,563	11,568
7	9,884	9,882	9,881	9,881	9,881	9,881	9,883	9,883	9,882	9,882
8	10,812	10,811	10,811	10,811	10,809	10,807	10,809	10,806	10,807	10,806
9	10,814	10,815	10,814	10,809	10,811	10,811	10,812	10,810	10,811	10,811
10	11,069	11,070	11,071	11,070	11,071	11,066	11,068	11,068	11,070	11,072
11	10,665	10,666	10,661	10,664	10,663	10,661	10,663	10,660	10,662	10,663
12	10,859	10,861	10,860	10,862	10,861	10,862	10,864	10,861	10,859	10,860
13	10,920	10,916	10,914	10,913	10,913	10,913	10,917	10,909	10,912	10,912
14	11,537	11,535	11,535	11,537	11,538	11,539	11,540	11,541	11,537	11,538
15	11,348	11,346	11,346	11,346	11,351	11,348	11,344	11,346	11,346	11,348
16	11,585	11,587	11,581	11,582	11,582	11,580	11,586	11,580	11,581	11,580
17	11,385	11,385	11,380	11,378	11,383	11,381	11,379	11,387	11,384	11,384
18	11,326	11,327	11,323	11,324	11,326	11,326	11,327	11,331	11,321	11,331
19	11,041	11,043	11,036	11,040	11,042	11,040	11,043	11,040	11,040	11,041
20	10,103	10,104	10,101	10,101	10,097	10,099	10,099	10,097	10,100	10,103
21	12,280	12,276	12,277	12,279	12,281	12,280	12,275	12,280	12,279	12,278
22	10,743	10,747	10,749	10,748	10,749	10,749	10,744	10,744	10,743	10,742
23	11,838	11,836	11,836	11,837	11,837	11,836	11,837	11,837	11,836	11,833
24	10,888	10,886	10,887	10,891	10,889	10,888	10,889	10,890	10,885	10,886
25	12,103	12,101	12,103	12,105	12,108	12,105	12,105	12,105	12,107	12,103
26	10,910	10,909	10,912	10,913	10,912	10,914	10,913	10,913	10,914	10,914
27	11,507	11,510	11,510	11,509	11,510	11,504	11,505	11,505	11,506	11,508
28	10,705	10,708	10,708	10,705	10,707	10,708	10,705	10,706	10,706	10,706
29	10,306	10,310	10,307	10,307	10,305	10,307	10,307	10,304	10,307	10,306
30	11,258	11,260	11,261	11,255	11,256	11,257	11,257	11,253	11,255	11,257
31	11,114	11,110	11,110	11,112	11,115	11,110	11,111	11,111	11,113	11,112
32	11,042	11,041	11,040	11,037	11,038	11,038	11,038	11,037	11,037	11,038
33	10,748	10,748	10,745	10,746	10,745	10,747	10,747	10,745	10,745	10,746

Nesse exemplo, inicia-se com uma frequência de *clock* de 12,026 MHz e a chave  $ch_1$  no nível 0. Portanto, o  $led_1$  deve estar apagado. A próxima etapa é ligar a chave  $ch_1$  ao nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 1 MHz, ou seja, ajustar o gerador de funções para 11,026 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, devemos repetir a operação anterior, ou seja, ligar a chave  $ch_1$  ao nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 1 MHz, ou seja, ajustar o gerador de funções para 10,026 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  acendeu, já se sabe que o valor da dezena e da unidade faz parte da resposta procurada.

Para descobrir o primeiro dígito após a vírgula coloca-se a chave  $ch_1$  no nível 1 para zerar os flip-flops, e aumenta-se a frequência do gerador de funções de 1 MHz, ou seja, ajustamos o gerador de funções para 11,026 MHz e coloca-se a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, deve-se ligar a chave  $ch_1$  no nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 0,1 MHz, ou seja, ajustar o gerador de funções para 10,926 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, deve-se ligar a chave  $ch_1$  ao nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 0,1 MHz, ou seja, ajustar o gerador de funções para 10,826 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, deve-se repetir essa última etapa até ajustar o gerador de funções em 10,026 MHz quando o  $led_1$  deve acender e se tornar conhecido o valor da primeira dígito após a vírgula.

Para descobrir o segundo dígito após a vírgula coloca-se a chave  $ch_1$  no nível 1 para zerar os flip-flops, e aumenta-se a frequência do gerador de funções de 0,1 MHz, ou seja, ajusta-se o gerador de funções para 10,126 MHz e coloca-se a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, devemos ligar a

chave  $ch_1$  no nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 0,01 MHz, ou seja, ajustar o gerador de funções para 10,116 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, devemos ligar a chave  $ch_1$  no nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 0,01 MHz, ou seja, ajustar o gerador de funções para 10,106 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, devemos repetir essa última etapa até ajustar o gerador de funções em 10,046 MHz, quando o  $led_1$  deve acender sem definir o valor da segunda casa após a vírgula.

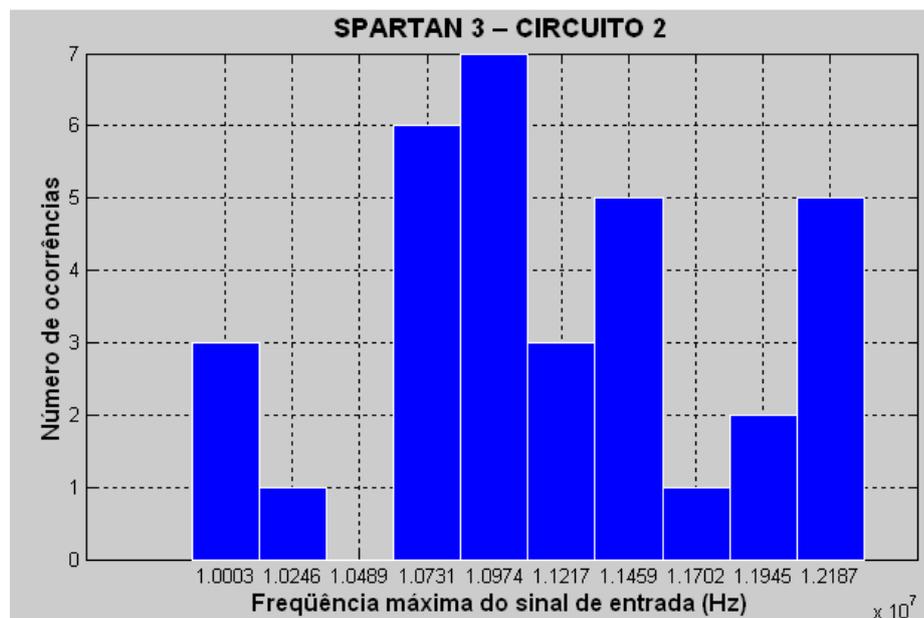
Para descobrir o terceiro dígito após a vírgula, coloca-se a chave  $ch_1$  no nível 1 para zerar os flip-flops, e aumenta-se a frequência do gerador de funções de 0,01 MHz, ou seja, ajusta-se o gerador de funções para 10,056 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, deve-se ligar a chave  $ch_1$  no nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 0,001 MHz, ou seja, ajustar o gerador de funções para 10,055 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  não acendeu, deve-se ligar a chave  $ch_1$  ao nível 1 para zerar os flip-flops, diminuir a frequência do gerador de funções de 0,001 MHz, ou seja, ajustar o gerador de funções para 10,054 MHz e colocar a chave  $ch_1$  no nível 0 para verificar se o  $led_1$  acende com essa frequência. Como o  $led_1$  acendeu, fica determinado o valor da segunda casa após a vírgula. Para determinar com certeza o valor da terceira casa após a vírgula é necessário repetir o mesmo processo com o quarto dígito após a vírgula e, se necessário, arredondar o valor da terceira casa após a vírgula.

Após a obtenção do valor da frequência do *clock* que faz o circuito 2 acender o  $led_1$  é necessário repetir o teste 10 vezes para cada área. Para isso, antes de cada teste o último valor obtido foi aumentado de 0,04 ou um múltiplo de 0,04. O múltiplo de 0,04 foi usado quando

o led<sub>1</sub> permaneceu aceso mesmo com o aumento da frequência do gerador de funções. Após isso, o valor da frequência do gerador de funções foi diminuída de 0,0001 MHz até o led<sub>1</sub> acender. A resposta do valor da frequência do gerador de funções foi obtida com precisão até a terceira casa decimal.

Após a obtenção dos dados da tabela 5.5 foi obtido o valor médio da frequência ( $f$ ) do sinal de *clock* de cada área individual. A figura 5.24 apresenta o número de ocorrências dos valores médios de cada área em relação à frequência máxima do sinal de *clock* do circuito 2. Usando os dados da média de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{f} = \frac{\sum_{i=1}^n f_i}{n} = 11,184 \text{ MHz}$$



**Figura 5.24 – Número de ocorrências dos valores médios medidos de cada área em relação à frequência máxima do sinal de *clock* do circuito 2 no FPGA 2XC3S200.**

Comparando o resultado mais afastado do valor médio de todo o conjunto de dados em relação ao valor médio de todo o conjunto de dados, observa-se que a máxima variação desse sinal em relação a média é de 11,64%.

Usando os valores médios de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (f_i - \bar{f})^2}{n - 1}} = 0,652 \text{ MHz}$$

Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $11,184 \text{ MHz} \pm 1,998\sigma$ .

Para calcular o valor médio do tempo de atraso ( $t_p$ ) de cada porta lógica de uma área individual foi utilizado o valor médio da frequência ( $f$ ) do sinal de *clock* de cada área da tabela 5.5 e a fórmula 3.4 do circuito 2. Usando os dados da média de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{t}_p = \frac{\sum_{i=1}^n t_{pi}}{n} = 0,808 \text{ ns}$$

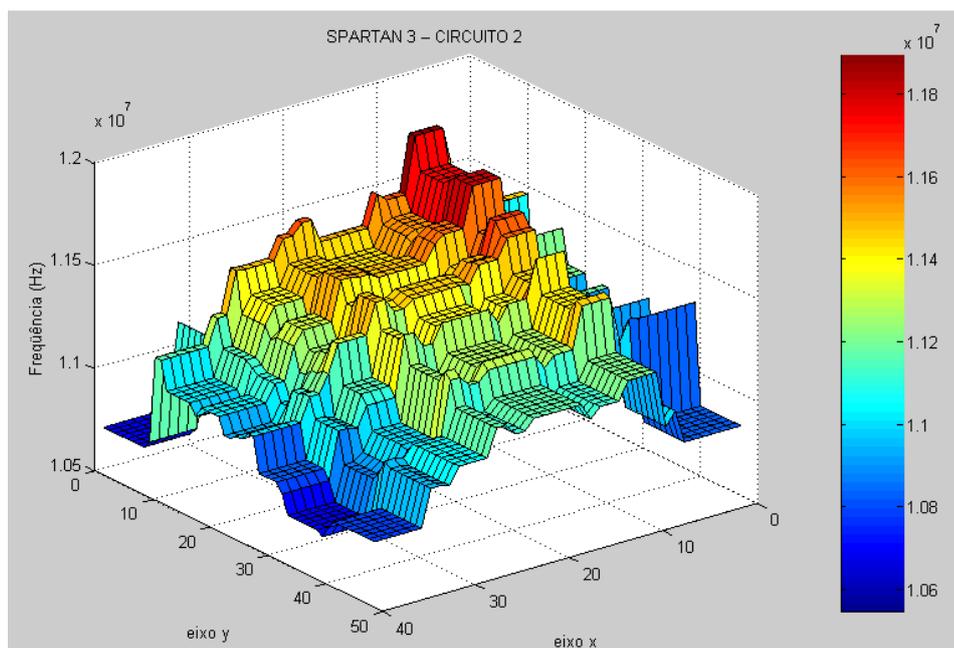
Comparando o resultado mais afastado do valor médio de todo o conjunto de dados em relação ao valor médio de todo o conjunto de dados, observa-se que a máxima variação desse sinal em relação à média é de 12,80%.

Usando os valores médios de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (t_{pi} - \bar{t}_p)^2}{n - 1}} = 0,047 \text{ ns}$$

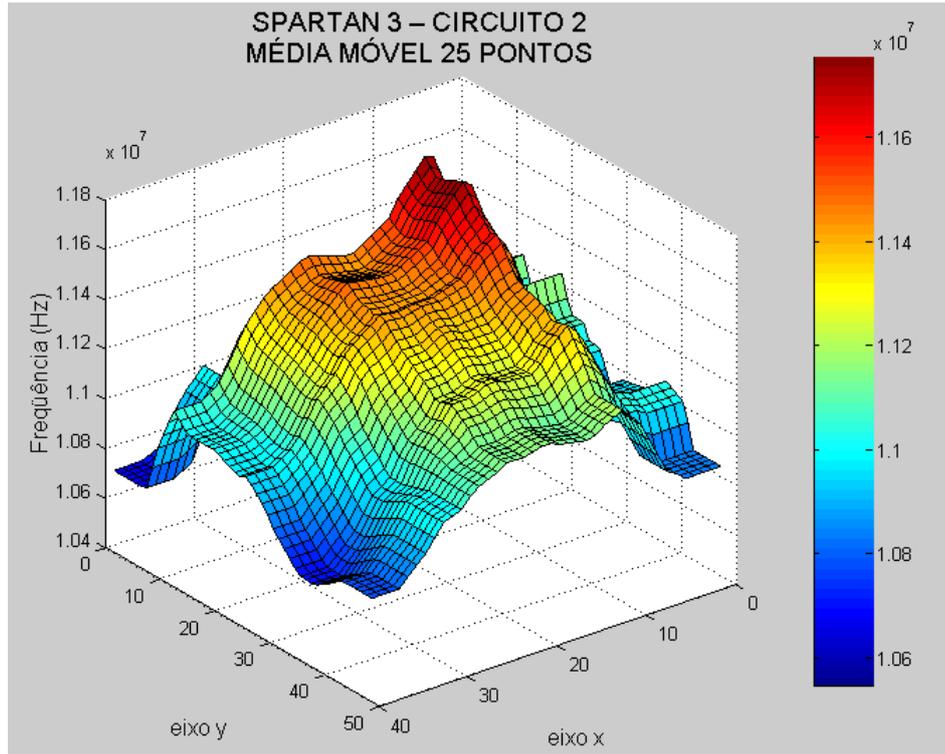
Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $0,808 \text{ ns} \pm 2,189\sigma$ .

A figura 5.25 apresenta o valor médio da frequência do sinal de *clock* obtido em cada CLB do circuito 2. As figuras C3 e C4 do apêndice C são repetições da figura 5.25 com o observador posicionado em outro local. Analisando essa figura, observa-se que existe uma variação espacial e uma variação local aleatória dos dados obtidos. Como o circuito 2 mede a frequência do caminho sobre teste que tem o maior atraso de propagação, conclui-se que nas áreas com uma frequência maior apresentam um atraso de propagação menor e que nas áreas com frequência menor, apresentam o atraso de propagação maior. Também observa-se que o atraso de propagação aumenta quando a coordenada *y* for menor do que 9 e maior do que 12. O atraso de propagação é mínimo na região em que *x* varia de 9 a 12 e *y* varia de 9 a 10 e é máximo na região em que *x* varia de 0 a 6 e *y* varia de 0 a 7.



**Figura 5.25 – Frequência do sinal de *clock* do circuito 2 associada às coordenadas do FPGA 2XC3S200.**

A figura 5.26 apresenta o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *clock* obtido em cada CLB do circuito 2. Nessa figura, observa-se que o método atenuou as flutuações da figura 5.25 e salientou as tendências de variação espacial do circuito.



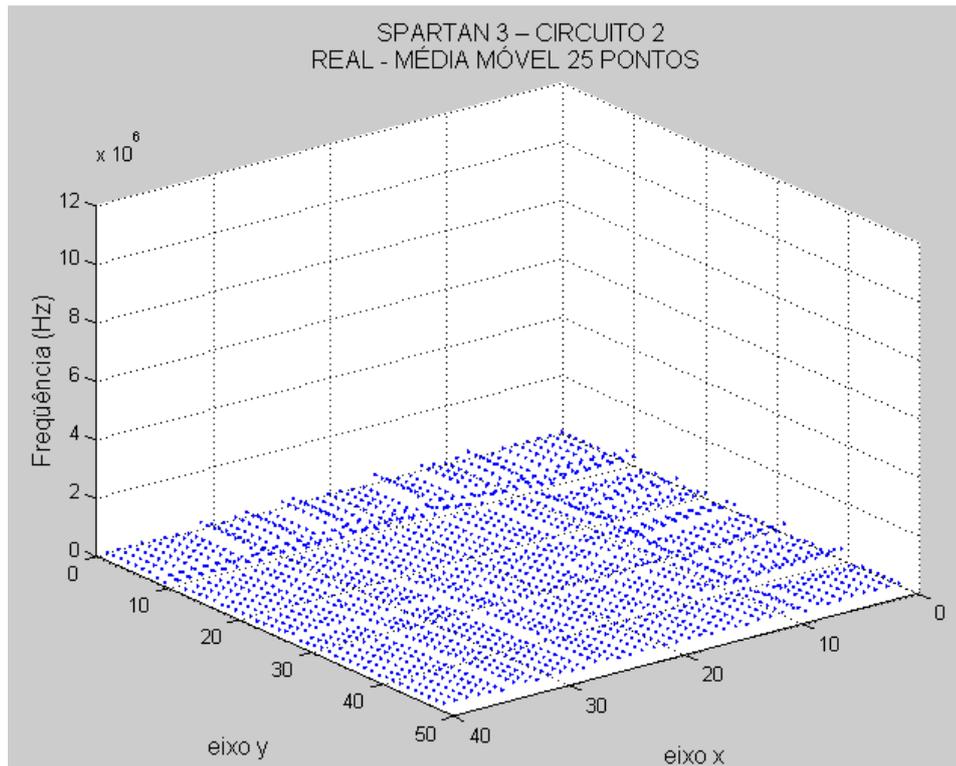
**Figura 5.26 – Média móvel dos dados do circuito 2 do FPGA 2XC3S200.**

A figura 5.27 apresenta o resíduo da comparação das figuras 5.25 e 5.26. Ou seja, a diferença do valor médio da frequência do sinal de *clock* obtido em cada CLB (figura 5.25) e o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *clock* obtido em cada CLB (figura 5.26) do circuito 2. Nessa figura, observa-se que o maior valor de resíduo equivale a 3,55% do maior valor medido e 4% do menor valor medido. Portanto, nesse circuito o método da média móvel de ordem 25 mostra claramente a correlação espacial do circuito.

Usando o valor médio da frequência do sinal de *clock* obtido em cada CLB do circuito 2, o sistema de equações E.4 e a equação E.3 do apêndice E, tem-se que o plano de mínimos quadrados que se ajusta ao conjunto de valores é:

$$Z = 1.10^{+7} ( 1,118 - 0,0004.X + 0,0001.Y ) \quad (5.3)$$

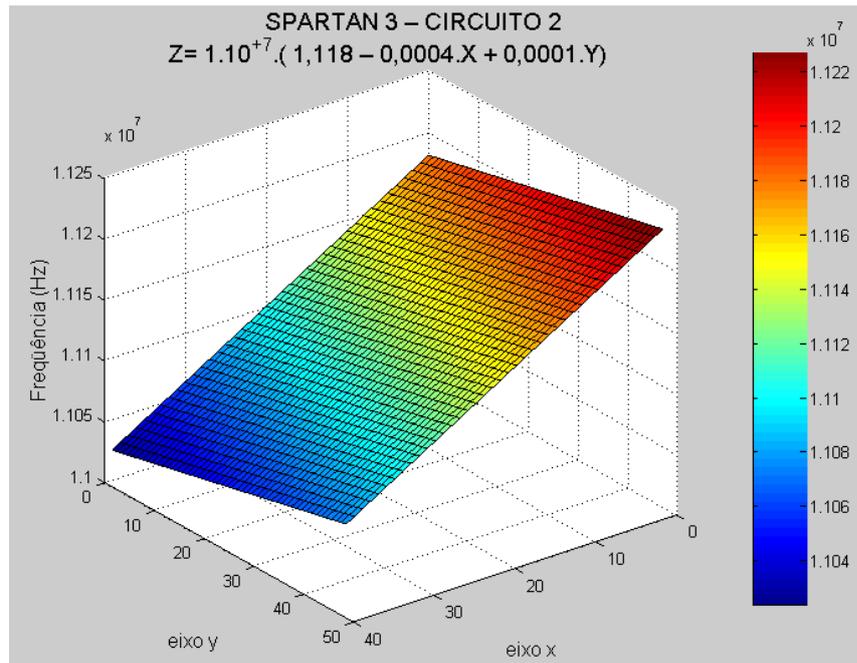
Nesta equação, o valor de  $Z$  representa o valor da frequência do sinal de *clock* obtido em cada CLB e os valores de  $X$  e  $Y$  representam as coordenadas de cada CLB.



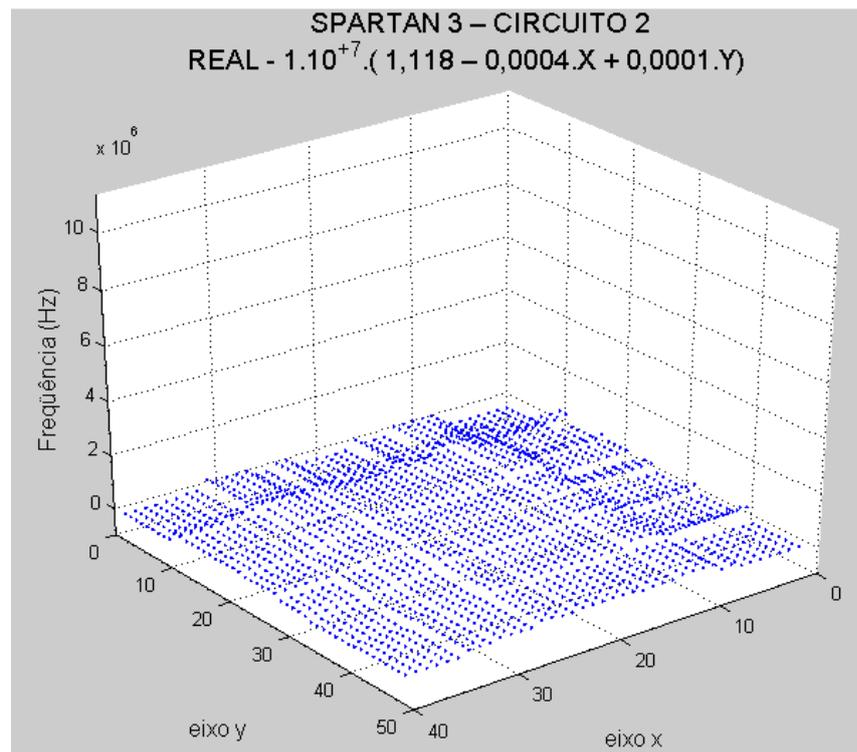
**Figura 5.27 – Diferença entre o conjunto de dados e a média móvel do circuito 2 do FPGA 2XC3S200.**

O plano de mínimos quadrados da equação 5.3, desenhado na figura 5.28, mostra a tendência da variação espacial do circuito.

A figura 5.29 apresenta o resíduo da comparação das figuras 5.25 e 5.28. Nessa figura, observa-se que o maior valor de resíduo equivale a 6,33% do maior valor medido e 7,14% do menor valor medido. Portanto, nesse circuito o método do plano de mínimos quadrados também mostra claramente a correlação espacial do circuito. Assim, tanto o método do plano de mínimos quadrados como a média móvel confirmam a forte correlação espacial no comportamento do circuito.

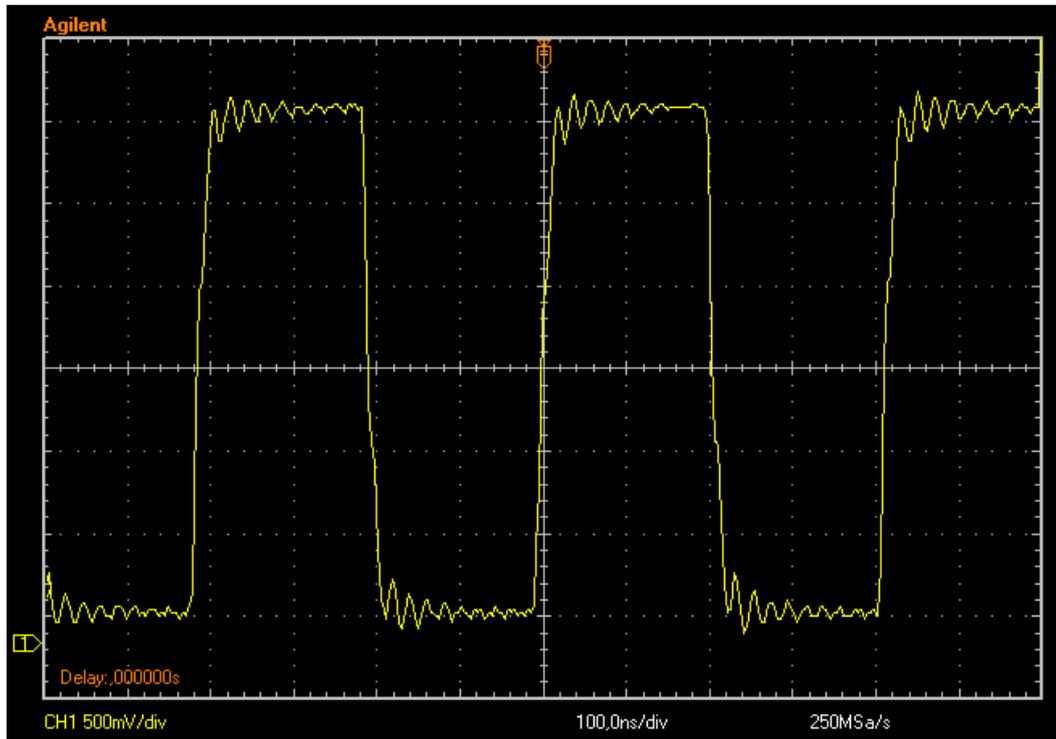


**Figura 5.28 – Plano de mínimos quadrados dos dados do circuito 2 do FPGA 2XC3S200.**



**Figura 5.29 – Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 2 do FPGA 2XC3S200.**





**Figura 5.32 – Forma de onda aplicada na entrada do circuito 2 implementado no FPGA EP2C35F672.**

A tabela 5.6 apresenta os resultados das medidas efetuadas no circuito da figura 5.30 para os dois caminhos sobre teste colocados nas 30 posições indicadas pela segunda, terceira e quarta coluna da tabela 4.2. Durante as medições, a temperatura ambiente variou de 25,0°C a 27,0°C.

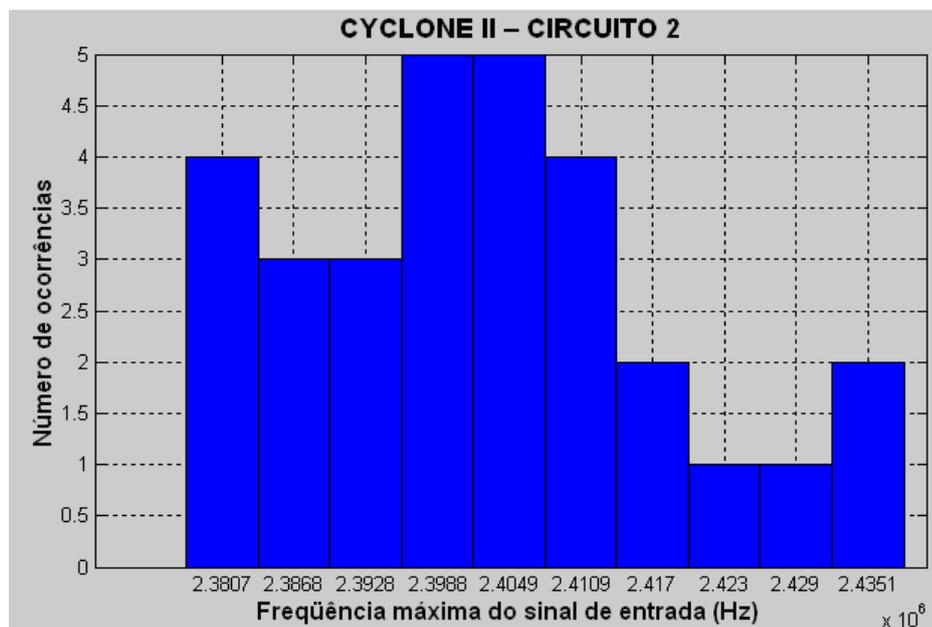
A rotina para obter cada resultado apresentado da tabela 5.6 é a mesma usada no circuito 2 no FPGA 2XC3S200.

Após a obtenção dos dados da tabela 5.6, obteve-se o valor médio da frequência ( $f$ ) do sinal de *clock* de cada área individual. A figura 5.33 apresenta o número de ocorrências dos valores médios de cada área em relação à frequência máxima do sinal de *clock* do circuito 2. Usando os dados das médias de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{f} = \frac{\sum_{i=1}^n f_i}{n} = 2,403 \text{ MHz}$$

Tabela 5.6 – Resultados das medidas efetuadas com o circuito 2 no FPGA EP2C35F672.

ÁREAS	ELEMENTOS DAS AMOSTRAS (MHz)									
	1	2	3	4	5	6	7	8	9	10
1	2,412	2,413	2,413	2,413	2,412	2,413	2,413	2,412	2,412	2,412
2	2,394	2,394	2,394	2,394	2,393	2,395	2,394	2,394	2,394	2,394
3	2,411	2,411	2,411	2,411	2,411	2,411	2,411	2,411	2,410	2,411
4	2,401	2,403	2,402	2,401	2,401	2,402	2,401	2,402	2,401	2,401
5	2,412	2,413	2,413	2,412	2,411	2,413	2,411	2,412	2,411	2,411
6	2,388	2,388	2,387	2,388	2,386	2,388	2,387	2,387	2,387	2,387
7	2,379	2,379	2,379	2,379	2,379	2,380	2,378	2,379	2,379	2,379
8	2,388	2,387	2,387	2,387	2,388	2,387	2,387	2,387	2,387	2,387
9	2,383	2,384	2,383	2,383	2,382	2,382	2,382	2,382	2,383	2,383
10	2,397	2,396	2,397	2,397	2,397	2,397	2,397	2,397	2,397	2,397
11	2,398	2,397	2,397	2,396	2,397	2,397	2,397	2,397	2,397	2,398
12	2,384	2,384	2,383	2,383	2,382	2,383	2,382	2,383	2,383	2,383
13	2,386	2,385	2,384	2,384	2,385	2,386	2,385	2,385	2,386	2,386
14	2,378	2,377	2,378	2,378	2,378	2,378	2,378	2,377	2,378	2,377
15	2,399	2,400	2,399	2,400	2,400	2,400	2,400	2,399	2,399	2,399
16	2,405	2,403	2,404	2,404	2,403	2,404	2,402	2,403	2,403	2,403
17	2,416	2,417	2,416	2,417	2,417	2,416	2,416	2,416	2,417	2,416
18	2,439	2,437	2,437	2,437	2,437	2,437	2,436	2,437	2,436	2,436
19	2,430	2,429	2,430	2,430	2,429	2,429	2,429	2,430	2,429	2,429
20	2,424	2,424	2,425	2,425	2,424	2,424	2,423	2,424	2,423	2,424
21	2,439	2,439	2,438	2,438	2,438	2,438	2,438	2,438	2,438	2,437
22	2,406	2,407	2,405	2,406	2,406	2,407	2,406	2,406	2,405	2,407
23	2,400	2,399	2,398	2,399	2,398	2,399	2,399	2,399	2,398	2,399
24	2,390	2,391	2,392	2,391	2,390	2,390	2,391	2,389	2,391	2,390
25	2,412	2,412	2,411	2,411	2,411	2,411	2,411	2,411	2,411	2,412
26	2,395	2,394	2,395	2,394	2,394	2,395	2,394	2,394	2,394	2,395
27	2,405	2,404	2,403	2,404	2,403	2,404	2,404	2,404	2,404	2,404
28	2,418	2,418	2,418	2,418	2,418	2,417	2,418	2,417	2,418	2,417
29	2,404	2,405	2,404	2,405	2,404	2,405	2,404	2,404	2,404	2,404
30	2,405	2,404	2,404	2,404	2,403	2,405	2,405	2,405	2,405	2,405

Figura 5.33 – Número de ocorrências dos valores médios de cada área em relação à frequência máxima do sinal de *clock* do circuito 2 no FPGA EP2C35F672.

Comparando o resultado mais afastado do valor médio de todo o conjunto de dados em relação ao valor médio de todo o conjunto de dados, observa-se que a máxima variação desse sinal em relação à média é de 1,47%.

Usando os valores médios de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (f_i - \bar{f})^2}{n - 1}} = 0,016 \text{ MHz}$$

Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de 2,403 MHz  $\pm$  2,19 $\sigma$ .

Para calcular o valor médio do tempo de atraso ( $t_p$ ) de cada porta lógica de uma área individual foi utilizado o valor médio da frequência ( $f$ ) do sinal de *clock* de cada área da tabela 5.6 e a fórmula 3.4 do circuito 2. Usando os dados da média de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{t}_p = \frac{\sum_{i=1}^n t_{pi}}{n} = 0,308 \text{ ns}$$

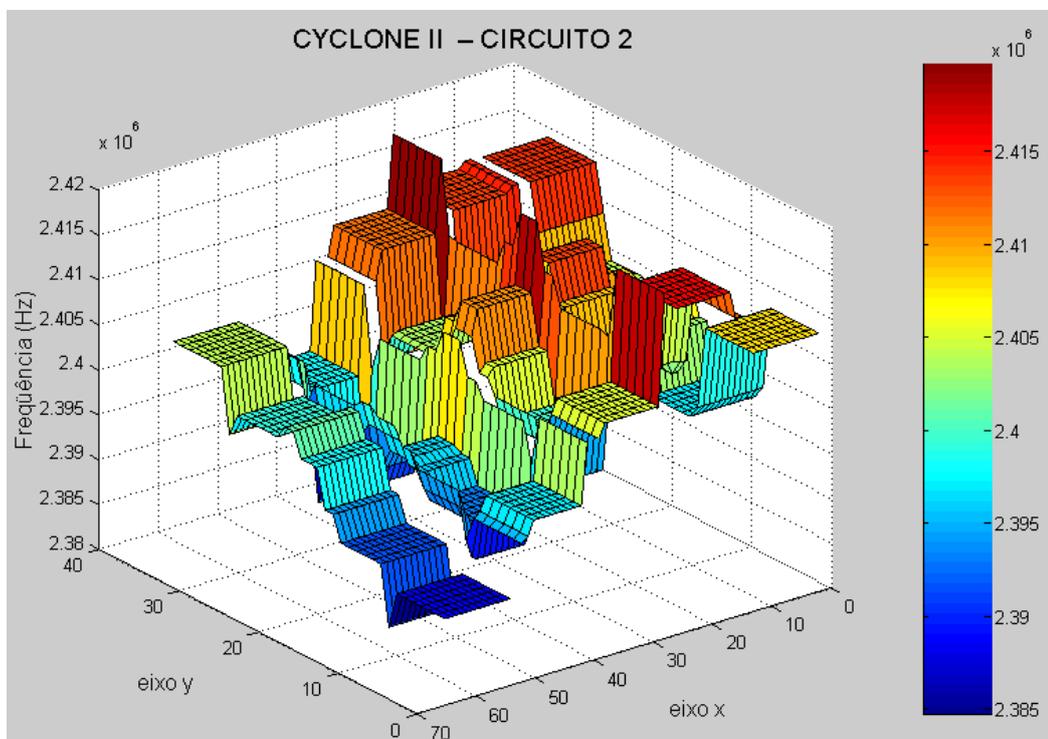
Comparando o resultado mais afastado do valor médio de todo o conjunto de dados em relação ao valor médio de todo o conjunto de dados, observa-se que a máxima variação desse sinal em relação à média é de 1,45%.

Usando os valores médios de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (t_{pi} - \bar{t}_p)^2}{n - 1}} = 2,053 \text{ ps}$$

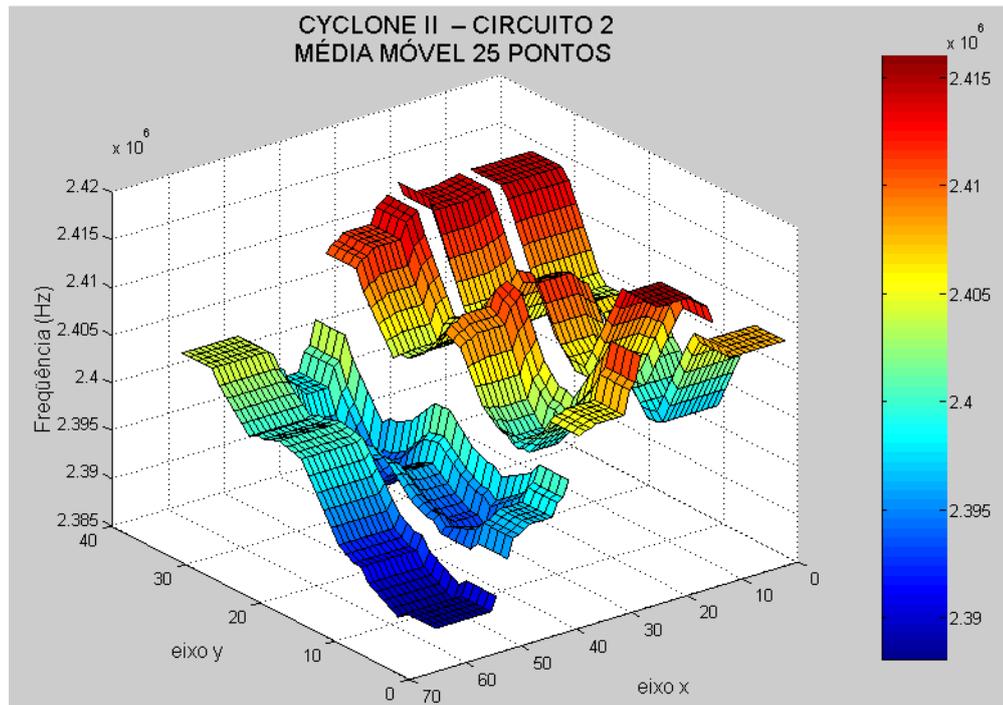
Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $0,308 \text{ ns} \pm 2,174\sigma$ .

A figura 5.34 apresenta o valor médio da frequência do sinal de *clock* obtido em cada LAB do circuito 2. A figura C8 do apêndice C é uma repetição da figura 5.34 com o observador posicionado em outro local. Analisando essa figura, observa-se que existe uma variação espacial e uma variação local aleatória dos dados obtidos. Como o circuito 2 mede a frequência do caminho sobre teste que tem o maior atraso de propagação, conclui-se que nas áreas com uma frequência maior apresentam um atraso de propagação menor e que nas áreas com frequência menor apresentam o atraso de propagação maior. De um modo geral, o atraso de propagação diminui conforme a área se afasta do cruzamento onde x varia de 53 a 64 e y é igual a 8. O atraso de propagação é mínimo na região em que x varia de 26 a 27 e y varia de 29 a 35 e é máximo na região em que x varia de 53 a 64 e y é igual a 8.



**Figura 5.34 – Frequência do sinal de *clock* do circuito 2 associada às coordenadas do FPGA EP2C35F672.**

A figura 5.35 apresenta o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *clock* obtido em cada LAB do circuito 2. Nessa figura, observa-se que o método atenuou as flutuações da figura 5.34 e salientou as tendências de variação espacial do circuito.



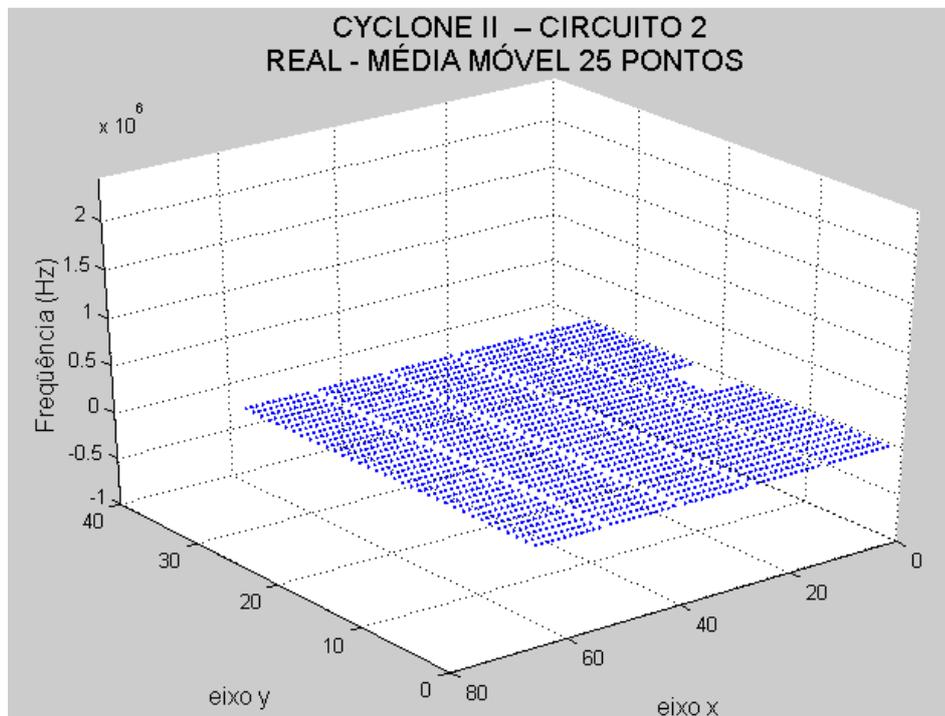
**Figura 5.35 – Média móvel dos dados do circuito 2 do FPGA EP2C35F672.**

A figura 5.36 apresenta o resíduo da comparação das figuras 5.34 e 5.35. Ou seja, a diferença do valor médio da frequência do sinal de *clock* obtido em cada LAB (figura 5.34) e o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *clock* obtido em cada LAB ((figura 5.35) do circuito 2. Nessa figura, observa-se que o maior valor de resíduo equivale a 0,42% do maior valor medido e 0,43% do menor valor medido. Portanto, nesse circuito o método da média móvel de ordem 25 demonstra claramente a correlação espacial do circuito.

Usando o valor médio da frequência do sinal de *clock* obtido em cada LAB do circuito 2, o sistema de equações E.4 e a equação E.3 do apêndice E, tem-se que o plano de mínimos quadrados que se ajusta ao conjunto de valores é:

$$Z = 1.10^{+6} \cdot (2,4072 - 0,0003 \cdot X + 0,0002 \cdot Y) \quad (5.4)$$

Nesta equação, o valor de *Z* representa o valor da frequência do sinal de *clock* obtido em cada LAB e os valores de *X* e *Y* representam as coordenadas de cada LAB.

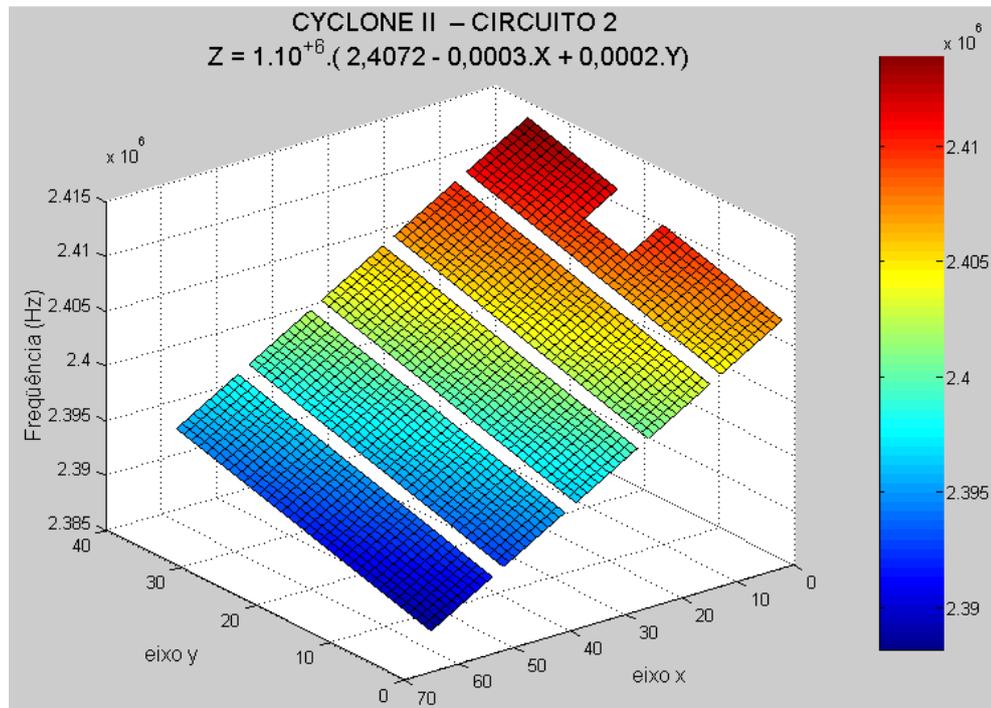


**Figura 5.36 – Diferença entre o conjunto de dados e a média móvel do circuito 2 do FPGA EP2C35F672.**

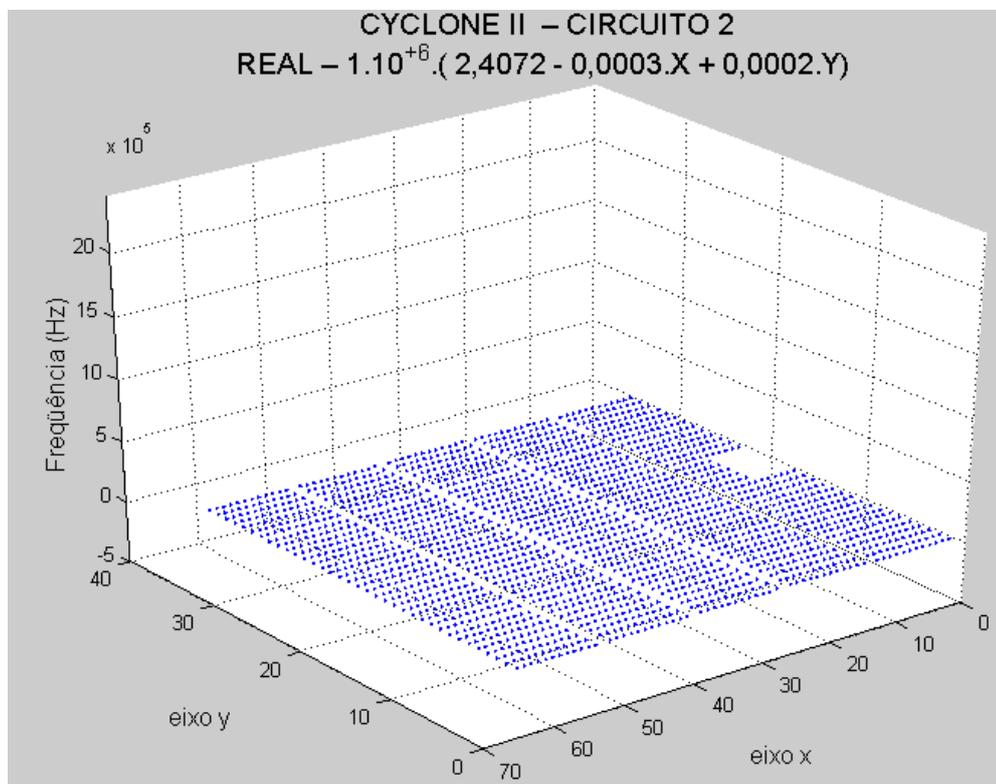
Na figura 5.37 construiu-se o plano de mínimos quadrados da equação 5.4, onde nota-se que o método mostra a tendência da variação espacial do circuito.

A figura 5.38 apresenta o resíduo da comparação das figuras 5.34 e 5.37. Nessa figura, observa-se que o maior valor de resíduo equivale a 0,76% do maior valor medido e 0,77% do menor valor medido. Permitindo assim, utilizar o gráfico do plano de mínimos quadrados para avaliar a correlação espacial do circuito. Portanto, nesse circuito o método do plano de mínimos quadrados também mostra claramente a correlação espacial do circuito. Assim,

tanto o método do plano de mínimos quadrados como a média móvel confirmam a forte correlação espacial no comportamento do circuito.



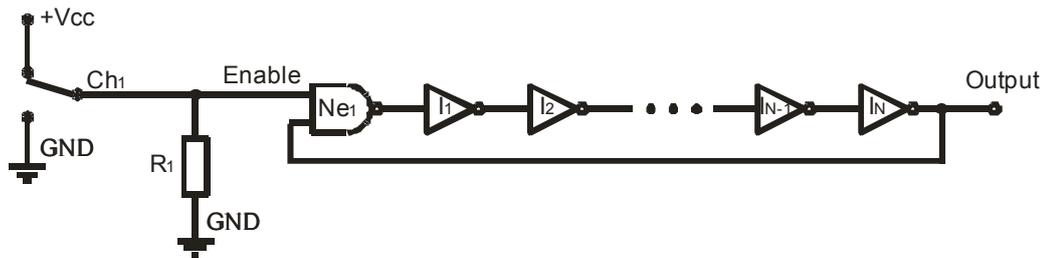
**Figura 5.37 – Plano de mínimos quadrados dos dados do circuito 2 do FPGA EP2C35F672.**



**Figura 5.38 – Diferença entre o conjunto de dados e o plano de mínimos quadrados do circuito 2 do FPGA EP2C35F672.**

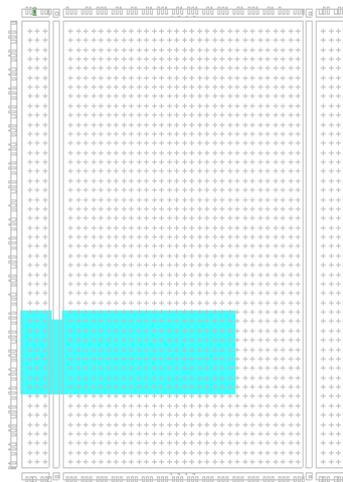
### 5.5 Circuito Oscilador em Anel Implementado no FPGA 2XC3S200

Para essa implementação foi utilizado o diagrama esquemático da fig. 5.39 com 110 inversores em série. O apêndice A.7 contém a descrição em VHDL do oscilador em anel.

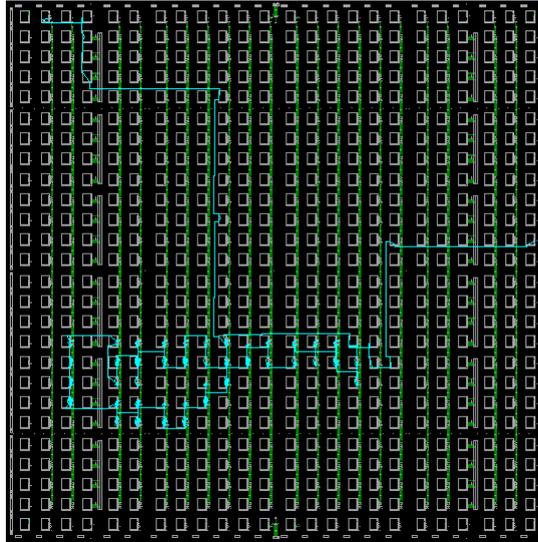


**Figura 5.39 – Diagrama esquemático do oscilador em anel prototipado na placa Spartan-3.**

A figura 5.40 apresenta a vista obtida com o programa “Floorplanner” após o “Place & Route” da área utilizada no FPGA pelo oscilador em anel colocado na posição 2 e a figura 4.41 apresenta a vista obtida com o programa “FPGA Editor” após o “Place & Route” do oscilador em anel colocado na posição 2 do FPGA.

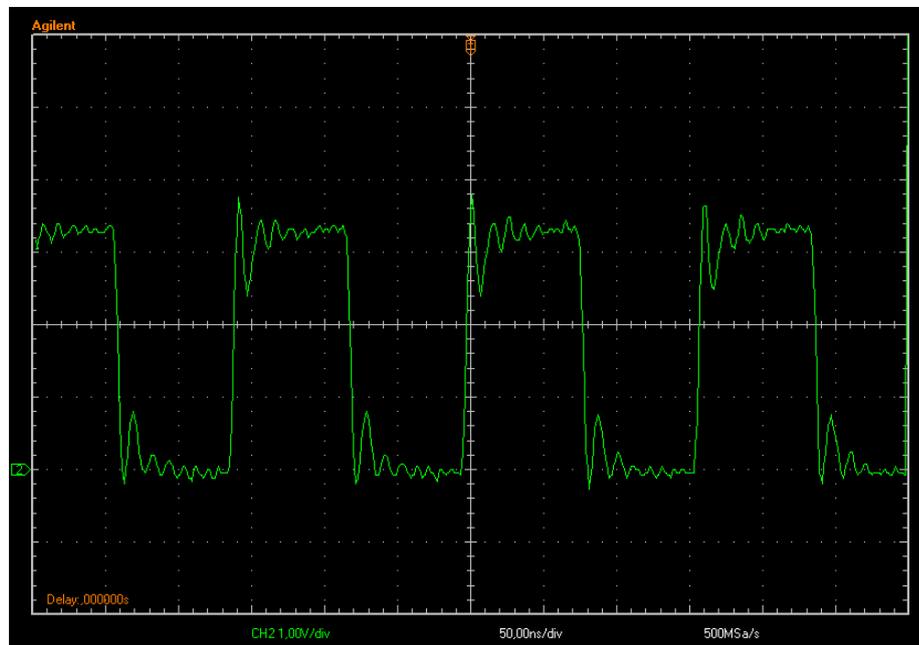


**Figura 5.40 – Vista da área disponível para o oscilador em anel colocado na posição 2 do FPGA XC3S200.**



**Figura 5.41 – Vista do oscilador em anel colocado na posição 2 do FPGA XC3S200.**

A figura 5.42 apresenta a forma de onda de saída do circuito oscilador em anel implementado na primeira posição do FPGA. As distorções mostradas nessa figura não foram corrigidas mesmo após a ponteira de prova ter sido ajustada conforme o manual do osciloscópio.



**Figura 5.42 – Forma de onda de saída do oscilador em anel colocado na primeira posição do FPGA XC3S200.**

A tabela 4.7 apresenta o período da tensão de saída do oscilador de anel colocado nas 33 posições indicadas pela segunda e terceira coluna da tabela 4.1. Durante as medições, a temperatura ambiente variou de 26,5°C a 27,5°C.

A rotina para obter cada resultado apresentado da tabela 5.7 consta do posicionamento da chave  $ch_1$  no nível 1 e da medição do período do sinal de saída com o osciloscópio. A chave  $ch_1$  ficou posicionada no nível 0 durante as posições.

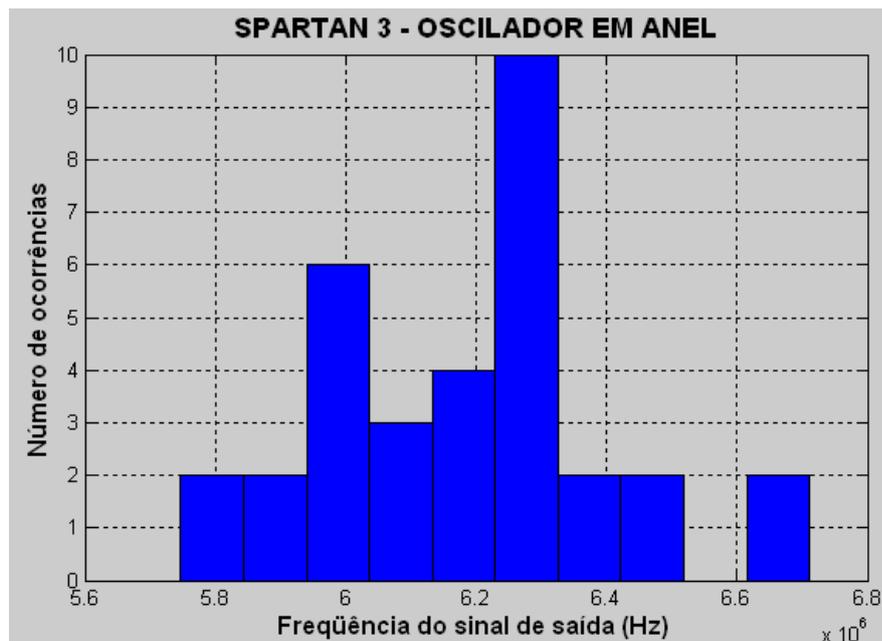
**Tabela 5.7 – Resultados do oscilador em anel no FPGA XC3S200.**

<b>Áreas</b>	<b>Elementos das amostras (ns)</b>
1	159,0
2	149,0
3	159,0
4	159,0
5	160,0
6	163,0
7	172,0
8	157,0
9	156,0
10	169,0
11	166,0
12	171,0
13	166,0
14	160,0
15	160,0
16	168,0
17	166,0
18	161,0
19	159,0
20	174,0
21	151,0
22	165,0
23	154,0
24	167,0
25	154,0
26	166,0
27	160,0
28	163,0
29	165,0
30	162,0
31	164,0
32	159,0
33	160,0

A partir dos dados da tabela 5.7 foi calculada a correspondente frequência da tensão de saída ( $f$ ) do oscilador de anel e construído o histograma da figura 5.43. Nesse histograma apresenta-se o número de ocorrências dos valores das áreas em relação à frequência da tensão de saída. Usando os dados de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{f} = \frac{\sum_{i=1}^n f_i}{n} = 6,183 \text{ MHz}$$

Comparando o resultado mais afastado do valor médio de todo o conjunto de dados em relação ao valor médio de todo o conjunto de dados, observa-se que a máxima variação desse sinal em relação à média é de 8,55%.



**Figura 5.43 – Número de ocorrências valores de cada área em relação à frequência do sinal de saída do oscilador em anel no FPGA 2XC3S200.**

Usando os valores da frequência de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}} = 0,222 \text{ MHz}$$

Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $6,183 \text{ MHz} \pm 2,38\sigma$ .

Se for aplicado o valor médio da frequência do sinal de saída do oscilador em anel e o número de inversores utilizados na fórmula 3.7 do oscilador em anel, tem-se que o valor médio do tempo de atraso de cada porta lógica é:

$$\bar{t}_p = \frac{1}{2 \cdot (110+1) \cdot 6,183 \cdot 10^6} = 0,729 \text{ ns}$$

Comparando o resultado mais afastado do valor médio de todo o conjunto de dados em relação ao valor médio de todo o conjunto de dados, observa-se que a máxima variação desse sinal em relação à média é de 7,99%.

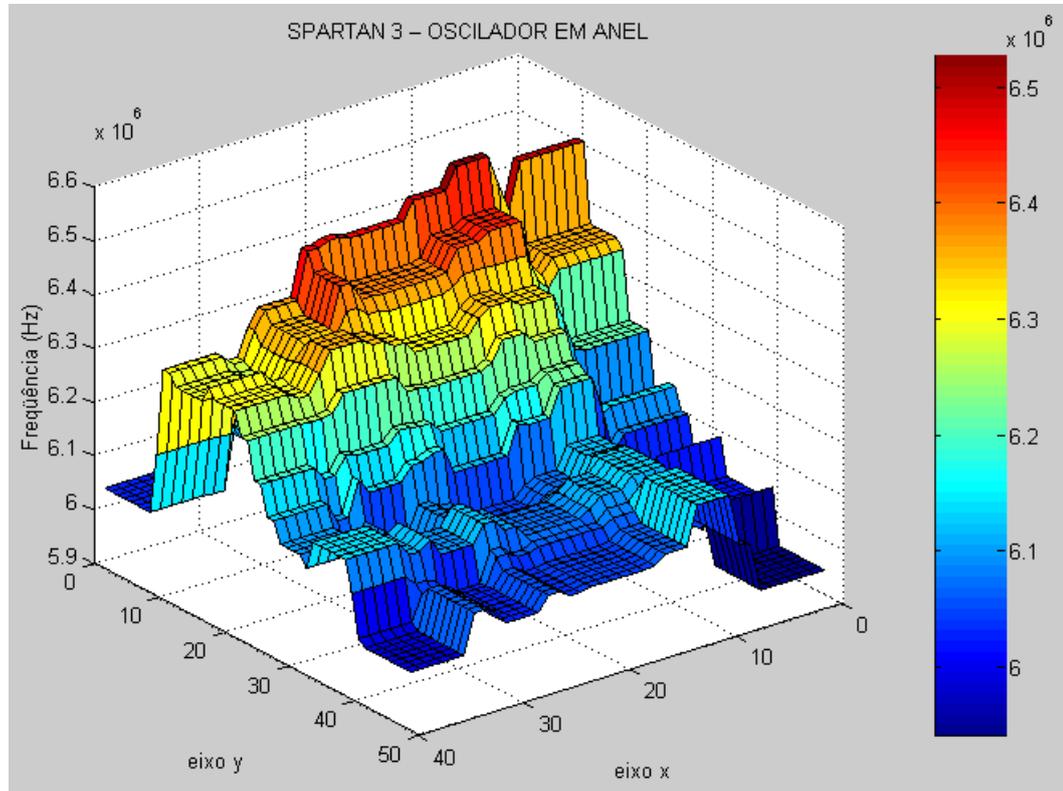
Com os dados de cada área da tabela 5.7 e a fórmula 3.7 do oscilador em anel foi calculado, para cada área, o tempo de atraso de cada porta lógica. Aplicando esses valores na fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (t_{pi} - \bar{t}_p)^2}{n - 1}} = 26,022 \text{ ps}$$

Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $0,729 \text{ ns} \pm 2,24\sigma$ .

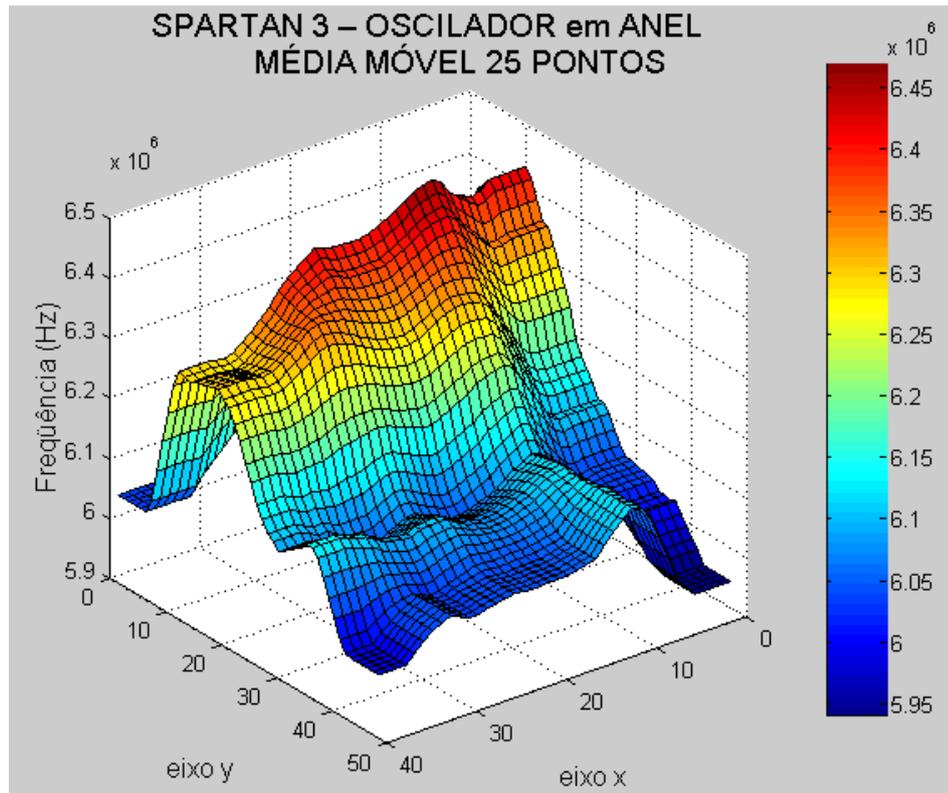
A figura 5.44 apresenta o valor médio da frequência do sinal de *output* obtidos em cada CLB do oscilador em anel. As figuras C5 e C6 do apêndice C são repetições da figura 5.44 com o observador posicionado em outro local. Analisando essa figura, nota-se que existe uma variação espacial e uma variação local aleatória dos dados obtidos. Como no oscilador em anel as áreas com frequência maior apresentam um atraso de propagação menor, conclui-se que o atraso de propagação é predominantemente menor nos CLB onde a coordenada y está

entre 9 e 10 e aumenta conforme afasta-se dela. O atraso de propagação é máximo nos CLB onde a coordenada x varia de 0 a 6 e a coordenada y varia de 40 a 47. O atraso de propagação é mínimo nos CLB onde a coordenada x varia de 9 a 12 e a coordenada y varia de 9 a 10.



**Figura 5.44 – Frequência do sinal de saída do oscilador em anel associado às coordenadas do FPGA 2XC3S200.**

A figura 5.45 apresenta o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *output* obtido em cada CLB do oscilador em anel. Nessa figura, observa-se que o método atenuou as flutuações da figura 5.44 e salientou as tendências de variação espacial do circuito.



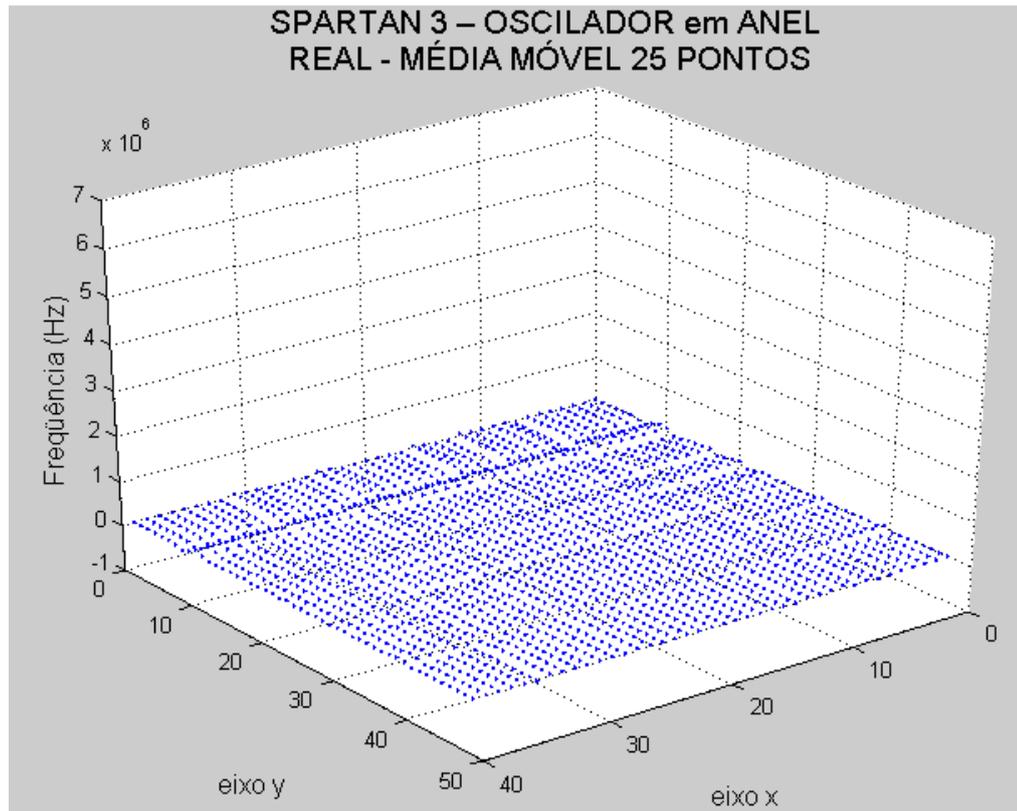
**Figura 5.45 – Média móvel dos dados do oscilador em anel do FPGA 2XC3S200.**

A figura 5.46 apresenta o resíduo da comparação das figuras 5.44 e 5.45. Ou seja, a diferença do valor médio da frequência do sinal de *output* obtido em cada CLB (figura 5.44) e o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *output* obtido em cada CLB (figura 5.45) do oscilador em anel. Nessa figura, observa-se que o maior valor de resíduo equivale a 2,15% do maior valor medido e 2,36% do menor valor medido. Portanto, nesse circuito o método da média móvel de ordem 25 mostra claramente a correlação espacial do circuito.

Usando o valor médio da frequência do sinal de *output* obtido em cada CLB do oscilador em anel, o sistema de equações E.4 e a equação E.3 do apêndice E, tem-se que o plano de mínimos quadrados que se ajusta ao conjunto de valores é:

$$Z = 1.10^{+6} \cdot (6,3434 - 0,0010 \cdot X - 0,0064 \cdot Y) \quad (5.5)$$

Nesta equação, o valor de Z representa o valor da frequência do sinal de *output* obtido em cada CLB e os valores de X e Y representam as coordenadas de cada CLB.



**Figura 5.46 – Diferença entre o conjunto de dados e a média móvel do oscilador em anel do FPGA 2XC3S200.**

Na figura 5.47 construiu-se o plano de mínimos quadrados da equação 5.5, onde observa-se que o método mostra a tendência da variação espacial do circuito.

A figura 5.48 apresenta o resíduo da comparação das figuras 5.44 e 5.47. Nessa figura, observa-se que o maior valor de resíduo equivale a 10,77% do maior valor medido e 11,84% do menor valor medido. Permitindo assim, utilizar o gráfico do plano de mínimos quadrados para avaliar a correlação espacial do circuito. Portanto, nesse circuito o método do plano de mínimos quadrados também mostra claramente a correlação espacial do circuito. Assim, tanto o método do plano de mínimos quadrados como a média móvel confirmam a forte correlação espacial no comportamento do circuito.

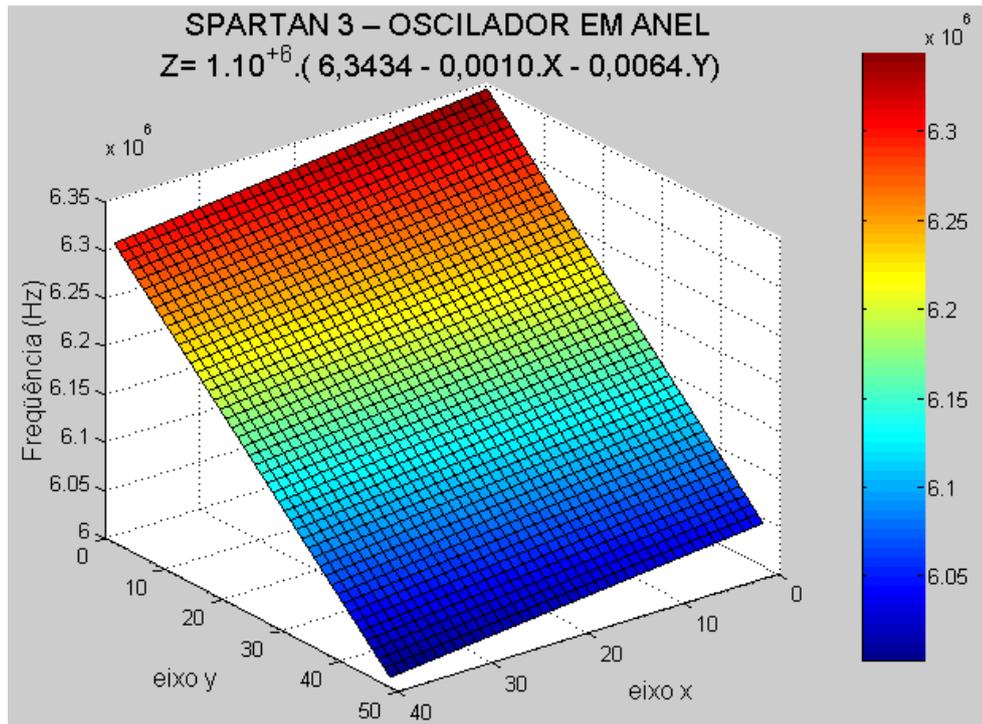


Figura 5.47 – Plano de mínimos quadrados dos dados do oscilador em anel do FPGA 2XC3S200.

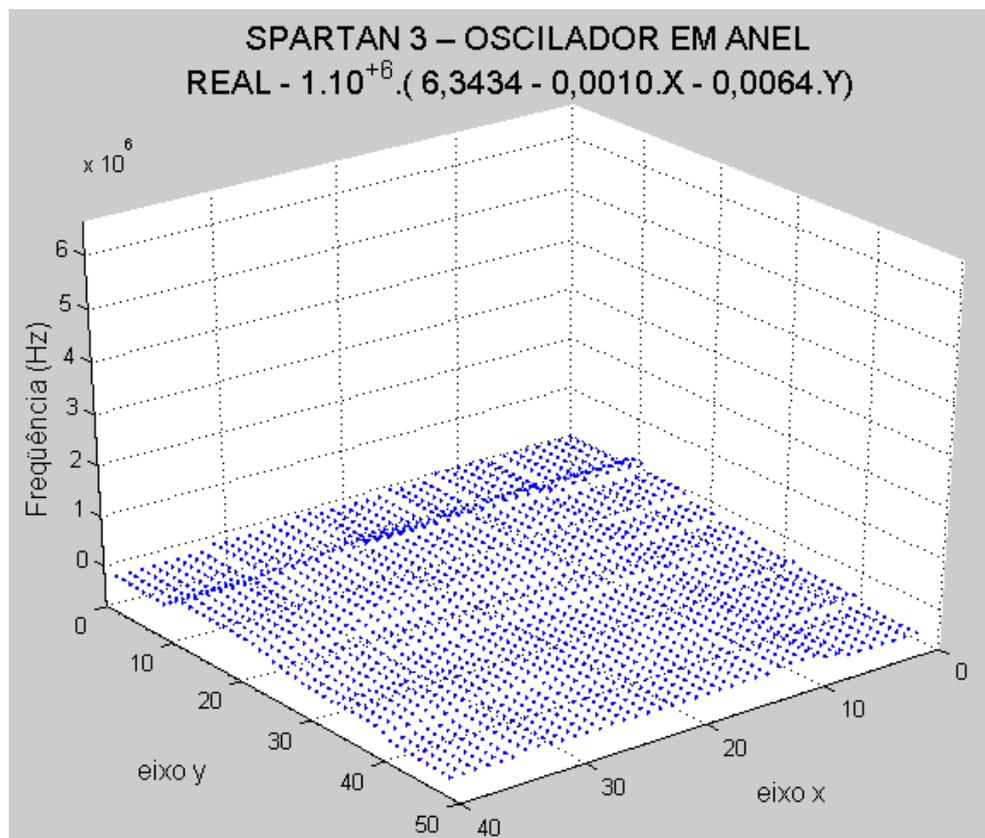
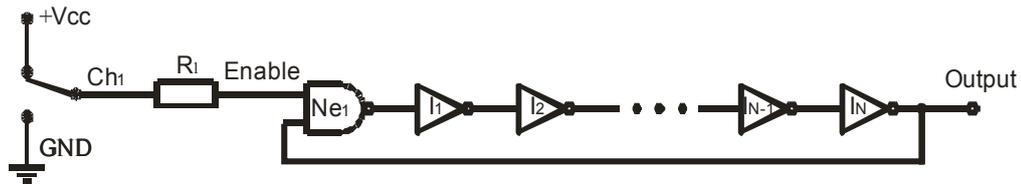


Figura 5.48 – Diferença entre o conjunto de dados e o plano de mínimos quadrados do oscilador em anel do FPGA 2XC3S200.

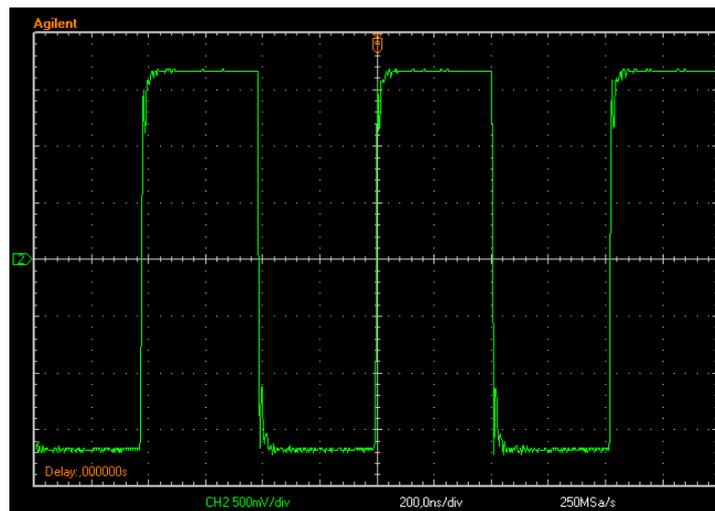
## 5.6 Circuito Oscilador em Anel Implementado no FPGA EP2C35F672

Para essa implementação foi utilizado o diagrama esquemático da figura 5.49 com 1350 inversores em série. O apêndice A.8 contém a descrição em VHDL do oscilador em anel.



**Figura 5.49 – Diagrama esquemático do oscilador em anel prototipado na placa Cyclone II.**

A figura 5.50 apresenta a forma de onda de saída do circuito oscilador em anel implementado na primeira posição do FPGA. As distorções mostradas nessa figura não foram corrigidas mesmo após a ponteira de prova ter sido ajustada conforme o manual do osciloscópio.



**Figura 5.50 – Forma de onda de saída do circuito oscilador em anel colocado na primeira posição do FPGA EP2C35F672.**

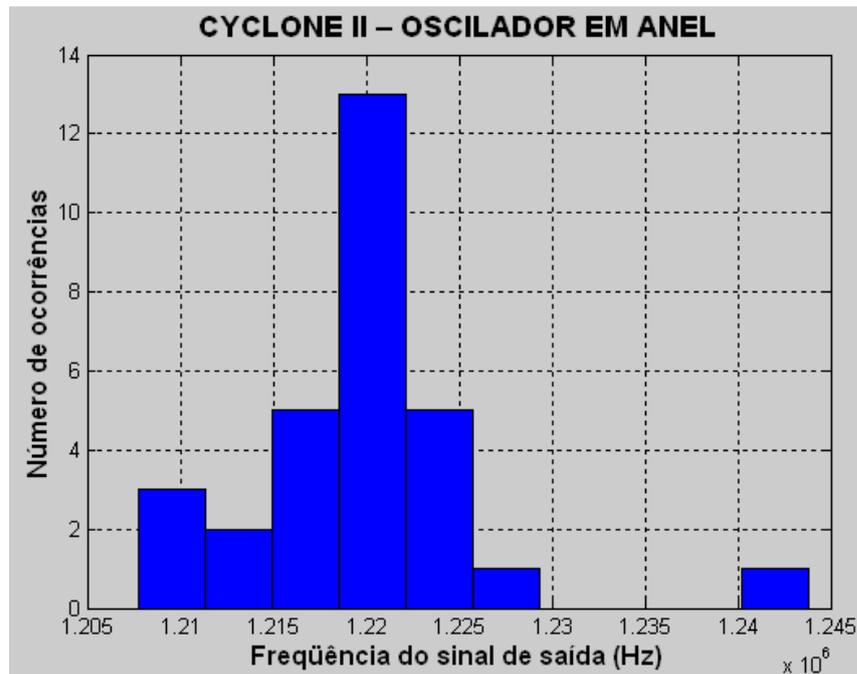
A tabela 5.8 apresenta o período da tensão de saída do oscilador de anel colocado nas 30 posições indicadas pela segunda, terceira e quarta colunas da tabela 4.2. Durante as medições a temperatura ambiente permaneceu em 26,0°C.

A rotina para obter cada resultado apresentado da tabela 5.8 é a mesma usada no oscilador em anel no FPGA 2XC3S200.

A partir dos dados da tabela 5.8 calculou-se a correspondente frequência da tensão de saída do oscilador de anel e construiu-se o histograma da figura 5.51. Neste histograma está representado o número de ocorrências dos valores das áreas em relação à frequência da tensão de saída.

**Tabela 5.8– Resultados do levantamento prático do oscilador em anel com 1350 inversores no FPGA EP2C35F672.**

<b>Áreas</b>	<b>Elementos das amostras (ns)</b>
<b>1</b>	820,0
<b>2</b>	820,0
<b>3</b>	824,0
<b>4</b>	820,0
<b>5</b>	816,0
<b>6</b>	820,0
<b>7</b>	822,0
<b>8</b>	824,0
<b>9</b>	820,0
<b>10</b>	814,0
<b>11</b>	822,0
<b>12</b>	820,0
<b>13</b>	826,0
<b>14</b>	820,0
<b>15</b>	816,0
<b>16</b>	820,0
<b>17</b>	804,0
<b>18</b>	818,0
<b>19</b>	818,0
<b>20</b>	822,0
<b>21</b>	820,0
<b>22</b>	820,0
<b>23</b>	816,0
<b>24</b>	828,0
<b>25</b>	820,0
<b>26</b>	820,0
<b>27</b>	822,0
<b>28</b>	828,0
<b>29</b>	820,0
<b>30</b>	822,0



**Figura 5.51 – Número de ocorrências dos valores das áreas em relação à frequência do sinal de saída do oscilador em anel no FPGA EP2C35F672.**

Usando os dados de cada área e a fórmula E.1 do apêndice E, tem-se que o valor médio de todo o conjunto de dados é:

$$\bar{f} = \frac{\sum_{i=1}^n f_i}{n} = 1,219 \text{ MHz}$$

Após, calculou-se o módulo da diferença entre o valor da frequência da tensão de saída mais afastado do valor médio de todo o conjunto de dados e o valor médio de todo o conjunto de dados. Em seguida, dividiu-se o valor encontrado pelo valor médio de todo o conjunto de dados e observou-se que a máxima variação desse sinal em relação à média foi de 2,00%.

Usando os valores da frequência de cada área, o valor médio de todo o conjunto de dados e a fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}} = 6,604 \text{ KHz}$$

Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $1,219 \text{ MHz} \pm 3,68\sigma$ .

Se for aplicado o valor médio da frequência do sinal de saída do oscilador em anel e o número de inversores utilizados na fórmula 3.7 do oscilador em anel, tem-se que o valor médio do tempo de atraso de cada porta lógica é:

$$\bar{t}_p = \frac{1}{2 \cdot (1350 + 1) \cdot 1,219 \cdot 10^6} = 0,304 \text{ ns}$$

Comparando o resultado mais afastado do valor médio de todo o conjunto de dados em relação ao valor médio de todo o conjunto de dados, observa-se que a máxima variação desse sinal em relação à média é de 1,96%.

Com os dados de cada área da tabela 5.8 e a fórmula 3.7 do oscilador em anel foi calculado, para cada área, o tempo de atraso de cada porta lógica. Aplicando esses valores na fórmula E.2 do apêndice E, tem-se que o desvio padrão resultante é:

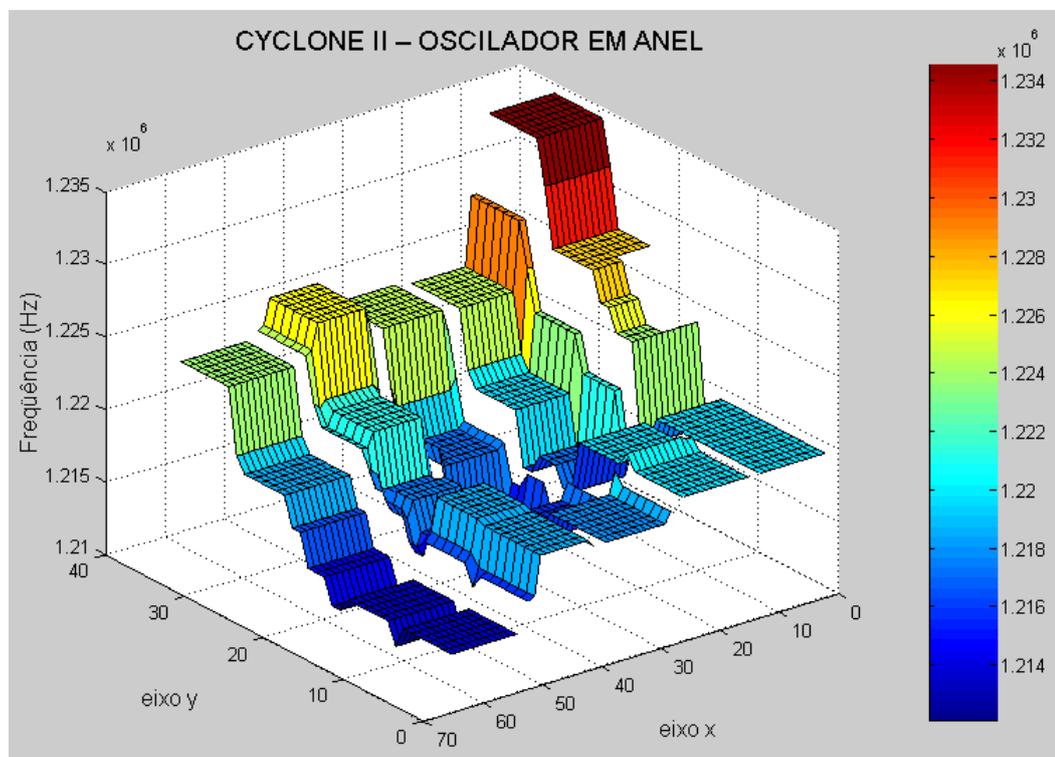
$$\sigma = \sqrt{\frac{\sum_{i=1}^n (t_{pi} - \bar{t}_p)^2}{n - 1}} = 1,632 \text{ ps}$$

Comparando o valor médio da área mais afastada do valor médio de todo o conjunto de dados em relação ao desvio padrão, pode-se dizer que os dados obtidos estão dentro da faixa de  $0,304 \text{ ns} \pm 3,64\sigma$ .

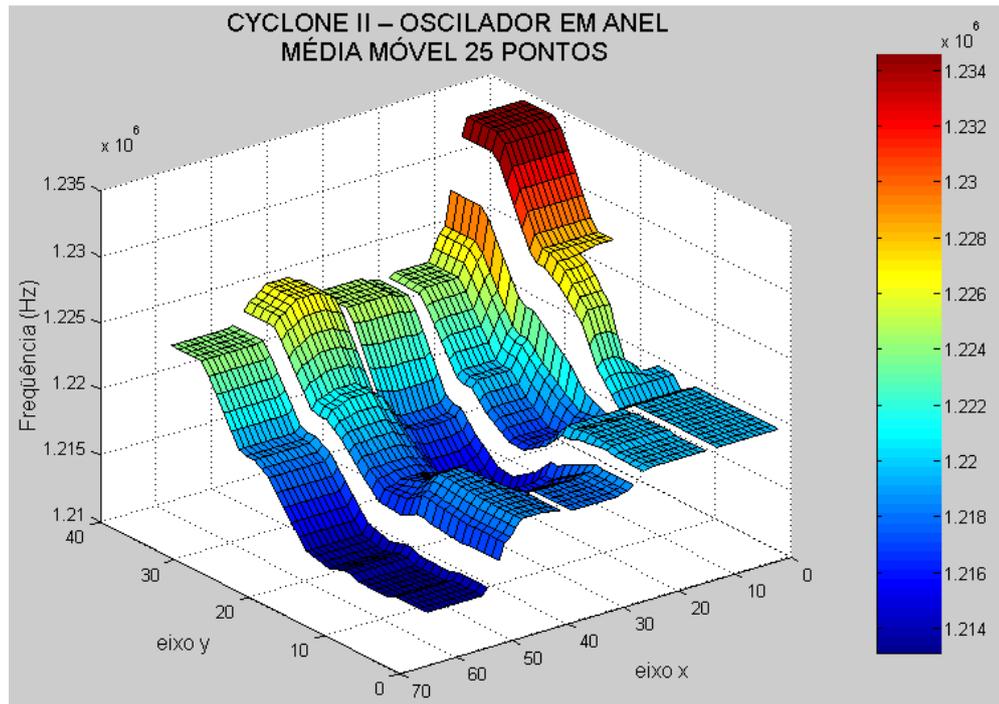
A figura 5.52 apresenta a média dos valores obtidos em cada LAB do oscilador em anel. A figura C9 do apêndice C é uma repetição da figura 5.52 com o observador posicionado em outro local. Analisando essa figura, observa-se que existe uma variação espacial e uma variação local aleatória dos dados obtidos. Como no oscilador em anel as áreas com frequência maior apresentam um atraso de propagação menor, conclui-se que o atraso de propagação é predominantemente menor nos CLB onde a coordenada y é igual a 35 ou a

coordenada x é igual a 1 e aumenta conforme se afastam dessas coordenadas. O atraso de propagação é máximo nos CLB onde a coordenada x varia de 53 a 64 e a coordenada y é igual a 8. O atraso de propagação é mínimo nos CLB onde a coordenada x varia de 1 a 12 e a coordenada y varia de 29 a 35.

A figura 5.53 apresenta o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *output* obtido em cada LAB do oscilador em anel. Nessa figura, observa-se que o método atenuou as flutuações da figura 5.52 e salientou as tendências de variação espacial do circuito.



**Figura 5.52 – Freqüência do sinal de saída do oscilador em anel associado às coordenadas do FPGA EP2C35F672.**



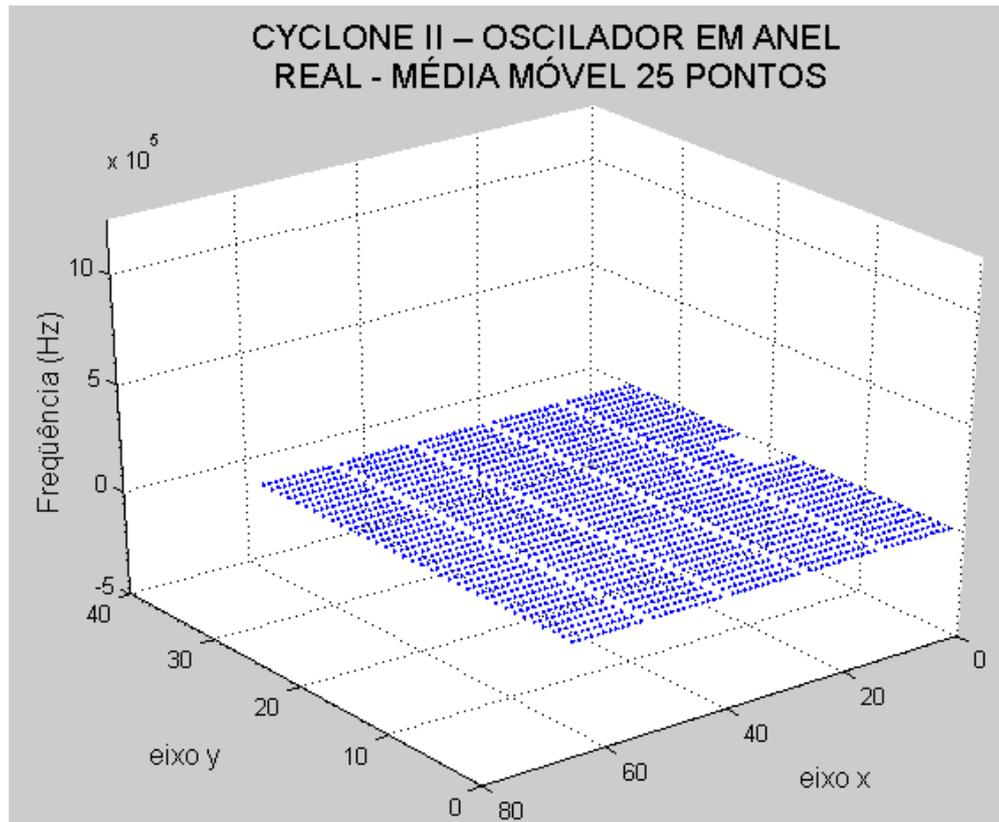
**Figura 5.53 – Média móvel dos dados do oscilador em anel do FPGA EP2C35F672.**

A figura 5.54 apresenta o resíduo da comparação das figuras 5.52 e 5.53. Ou seja, a diferença do valor médio da frequência do sinal de *output* obtido em cada LAB (figura 5.52) e o resultado da aplicação da média móvel de ordem 25 sobre o valor médio da frequência do sinal de *output* obtido em cada LAB ((figura 5.53) do circuito 2. Nessa figura, observa-se que o maior valor de resíduo equivale a 0,35% do maior valor medido e 0,36% do menor valor medido. Portanto, nesse circuito o método da média móvel de ordem 25 demonstra nitidamente a correlação espacial do circuito.

Usando o valor médio da frequência do sinal de *output* obtido em cada LAB do circuito 2, o sistema de equações E.4 e a equação E.3 do apêndice E, tem-se que o plano de mínimos quadrados que se ajusta ao conjunto de valores é:

$$Z = 1.10^{+6} \cdot (1,2184 - 0,0001 \cdot X + 0,0003 \cdot Y) \quad (5.6)$$

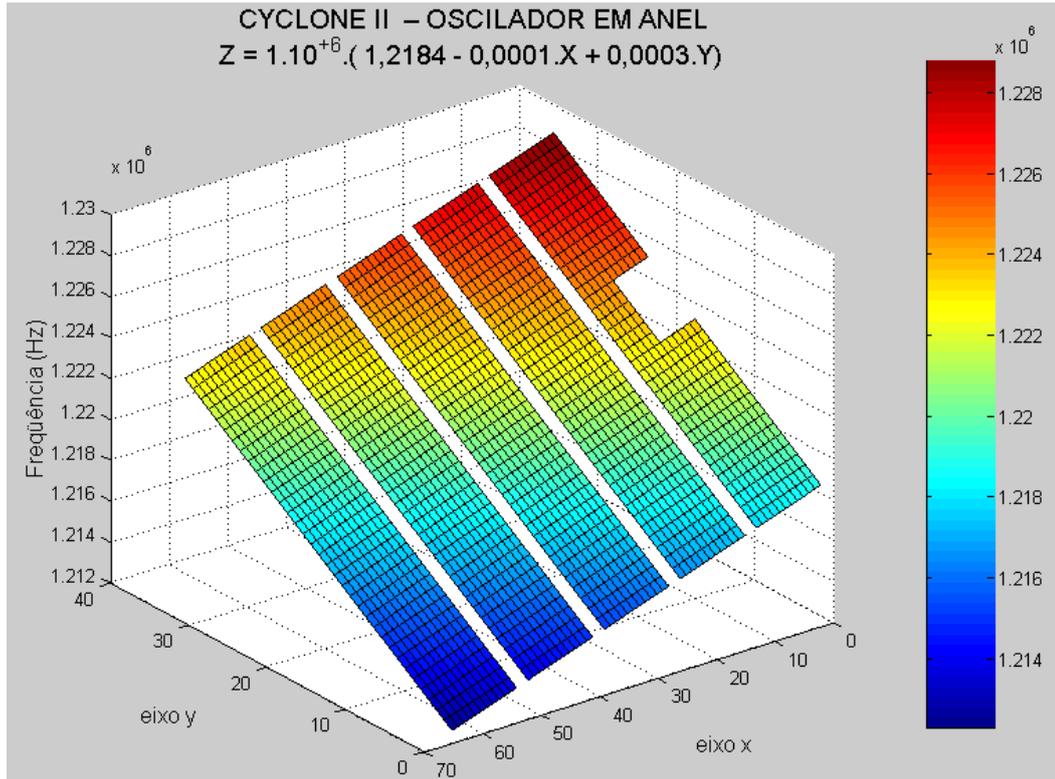
Nesta equação, o valor de  $Z$  representa o valor da frequência do sinal de *output* obtido em cada LAB e os valores de  $X$  e  $Y$  são as coordenadas de cada LAB.



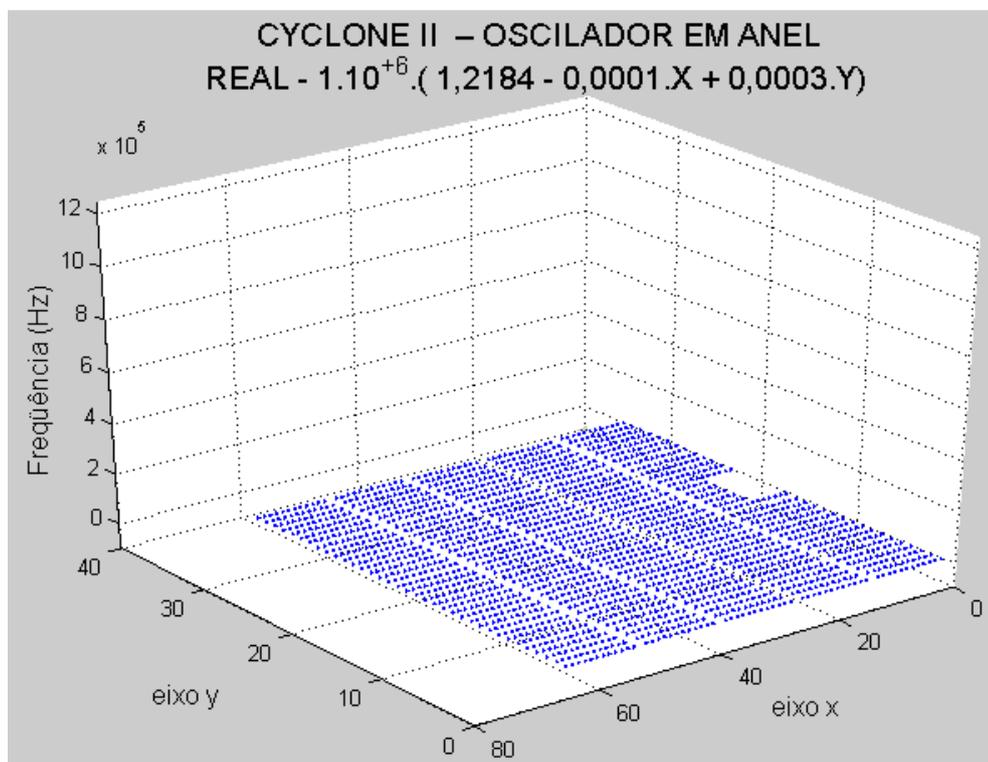
**Figura 5.54 – Diferença entre o conjunto de dados e a média móvel do oscilador em anel do FPGA EP2C35F672.**

Na figura 5.55 construiu-se o plano de mínimos quadrados da equação 5.6, onde nota-se que o método mostra a tendência da variação espacial do circuito.

A figura 5.56 apresenta o resíduo da comparação das figuras 5.52 e 5.55. Nessa figura, observa-se que o maior valor de resíduo equivale a 0,72% do maior valor medido e 0,73% do menor valor medido. Permitindo assim, utilizar o gráfico do plano de mínimos quadrados para avaliar a correlação espacial do circuito. Portanto, nesse circuito o método do plano de mínimos quadrados também mostra claramente a correlação espacial do circuito. Assim, tanto o método do plano de mínimos quadrados como a média móvel confirmam a forte correlação espacial no comportamento do circuito.



**Figura 5.55 – Plano de mínimos quadrados dos dados do oscilador em anel do FPGA EP2C35F672.**



**Figura 5.56 – Diferença entre o conjunto de dados e o plano de mínimos quadrados do oscilador em anel do FPGA EP2C35F672.**

## 5.7 Análise e Discussão dos Resultados

Independente do tipo de FPGA, a análise dos valores medidos, dos valores calculados, dos histogramas e dos gráficos 3D mostram a existência de variações *within-die*. Para o circuito 1, a componente randômica é mais acentuada se comparada com os circuitos 2 e oscilador em anel. O circuito 2 e o oscilador em anel apresentam uma correlação espacial mais acentuada.

Nas tabelas 5.9 e 5.10 apresenta-se um resumo dos cálculos efetuados na dissertação. Ao analisar os dados da relação  $\sigma/\mu$  da tabela 5.9, observa-se que nas duas placas os valores medidos apresentam tendência a estarem distribuídos próximos do valor médio nos circuitos 2 e oscilador em anel e tendência a não apresentar valores medidos distribuídos próximos do valor médio no circuito 1. Se for associado ao número de pulsos médio do sinal de saída dos circuitos 1 (tabela 5.9) ao valor do período da porta NE<sub>3</sub> do circuito 1 modificado, verifica-se que a placa Cyclone II apresenta uma diferença maior entre os tempos de atraso dos dois circuitos combinacionais sobre teste.

**Tabela 5.9 – Resumo dos cálculos dos valores medidos.**

	SPARTAN 3			CYCLONE II		
	Circuito 1	Circuito 2	Osc. Anel	Circuito 1	Circuito 2	Osc. Anel
<b>Média</b>	4,12 ciclos	11,18 MHz	6,18 MHz	27 ciclos	2,40 MHz	1,22 MHz
<b>Desvio padrão</b>	2,19 ciclos	0,65 MHz	0,22 MHz	4,47ciclos	0,02 MHz	6,60 KHz
<b>Relação entre desvio padrão e média (<math>\sigma/\mu</math>)</b>	53%	5,8%	3,6%	17%	0,83%	0,54%

A análise do circuito 1, usando a média móvel e do plano de mínimos quadrados, não mostraram uma clara correlação espacial no comportamento do circuito, mas apontaram a existência de uma grande componente randômica. Já nos circuitos 2 e oscilador em anel, eles confirmam uma forte correlação espacial no comportamento dos circuitos. Encontra-se nas

tabelas 5.11 e 5.12 um resumo das relações entre o valor do maior resíduo de cada circuito e o maior e menor valor medido de cada circuito.

**Tabela 5.10 – Resumo dos cálculos relacionados ao tempo de atraso de cada porta.**

	SPARTAN 3		CYCLONE II	
	Circuito 2	Oscilador em anel	Circuito 2	Oscilador em anel
<b>Média</b>	0,81 ns	0,73 ns	0,31 ns	0,30 ns
<b>Desvio padrão</b>	0,05 ns	26,02 os	2,05 ps	1,63 ps
<b>Relação entre desvio padrão e média (<math>\sigma/\mu</math>)</b>	6,2%	3,6%	0,7%	0,5%

**Tabela 5.11 – Resumo das relações dos resíduo com os valores medidos no Spartan 3.**

	MAIOR RESIDUO					
	Circuito 1		Circuito 2		Osc. Anel	
	MM	MQ	MM	MQ	MM	MQ
<b>Maior Valor Medido</b>	15,57%	28,57%	3,55%	6,33%	2,15%	10,77%
<b>Menor Valor Medido</b>			4%	7,14%	2,36%	11,84%

Nota: MM= Média Móvel

MQ= Mínimos Quadrados

**Tabela 5.12 – Resumo das relações dos resíduo com os valores medidos no Cyclone II.**

	MAIOR RESIDUO					
	Circuito 1		Circuito 2		Osc. Anel	
	MM	MQ	MM	MQ	MM	MQ
<b>Maior Valor Medido</b>	9,3%	16,13%	0,42%	0,76%	0,35%	0,72%
<b>Menor Valor Medido</b>			0,43%	0,77%	0,36%	0,73%

Nota: MM= Média Móvel

MQ= Mínimos Quadrados

Para cada FPGA, ao comparar as médias do circuito 2 e do oscilador em anel, utilizando o teste estatístico de hipótese para médias t-student, verifica-se que existe diferença significativa entre as médias desses circuitos com um intervalo de confiança de 95% ( $P < 0,05$ ) (figura 5.57). Apesar dessa diferença, nas figuras 5.58a e 5.58d para o Spartan 3 e nas figuras 5.59a e 5.59b para o Cyclone II, vê-se que: a variação de freqüência é pequena, as superfícies apresentam uma inclinação semelhante, os picos das ondulações diminuem conforme a superfície se torna mais baixa e os valores mínimos e máximos resultantes de testes diferentes estão localizados na mesma posição ou em outra cujo valor é

aproximadamente igual. As tabelas 5.13 e 5.14 apresentam a relação entre os tempos de atraso mínimo e máximo dos gráfico 3D para o circuito 2 e o oscilador em anel. Na tabela 5.13, o valor do  $t_{p \text{ máx}}$  do circuito 2 na posição  $0 \leq y \leq 7$  vale 94,8 ns e na posição  $40 \leq y \leq 47$  vale 92,3 ns. Na tabela 5.14, o valor do  $t_{p \text{ mín}}$  do circuito 2 na posição  $1 \leq x \leq 12$  vale 0,414  $\mu\text{s}$  e na posição  $26 \leq x \leq 27$  vale 0,413  $\mu\text{s}$ .

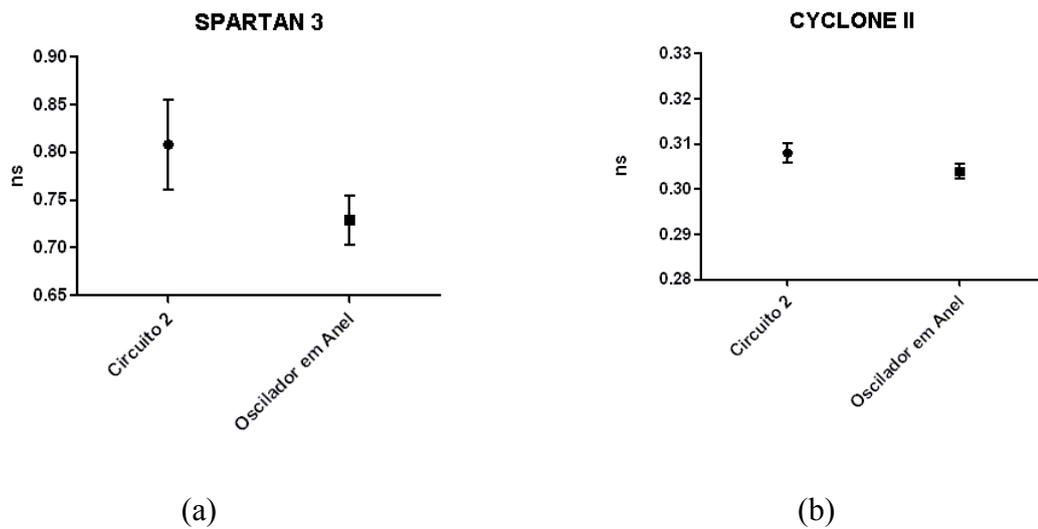


Figura 5.57 – Gráficos comparando o valor médio e o desvio padrão de cada porta lógica do circuito 2 e do oscilador em anel no (a) Spartan 3 e (b) no Cyclone II.

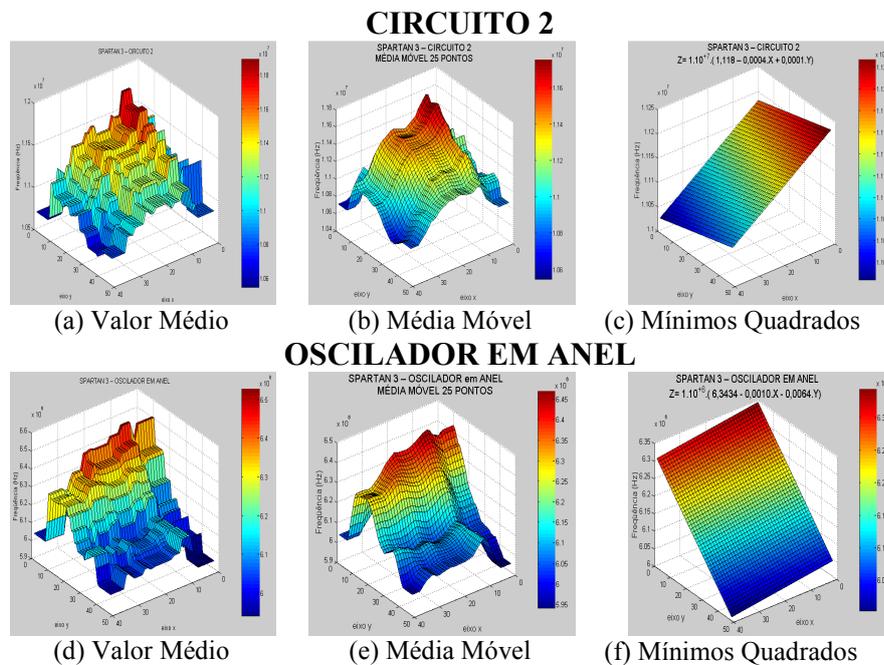


Figura 5.58 – Gráficos 3D do circuito 2 e oscilador em anel no Spartan 3.

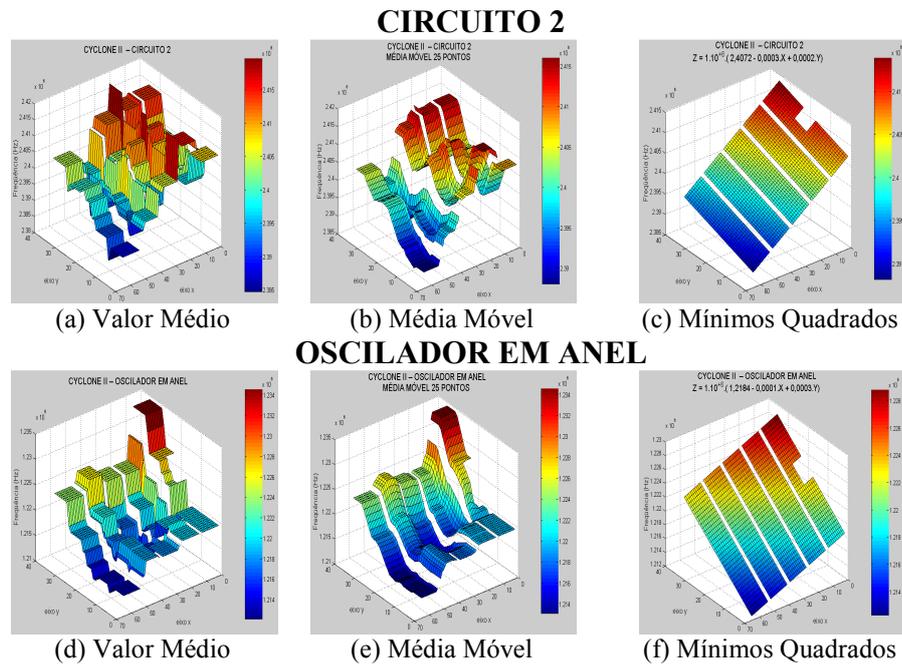


Figura 5.59 – Gráficos 3D do circuito 2 e oscilador em anel no Cyclone II.

Tabela 5.13– Análise do tempo de atraso do gráfico 3D no Spartan 3.

Circuito 2		Oscilador em anel	
$t_p \text{ mín}$	$t_p \text{ Max}$	$t_p \text{ mín}$	$t_p \text{ Max}$
$9 \leq x \leq 12$	$0 \leq x \leq 6$	$9 \leq x \leq 12$	$0 \leq x \leq 6$
$9 \leq y \leq 10$	$0 \leq y \leq 7$	$9 \leq y \leq 10$	$40 \leq y \leq 47$
(em $40 \leq y \leq 47$ é levemente inferior)			

Tabela 5.14 – Análise do tempo de atraso do gráfico 3D no Cyclone II.

Circuito 2		Oscilador em anel	
$t_p \text{ mín}$	$t_p \text{ máx}$	$t_p \text{ mín}$	$t_p \text{ Max}$
$26 \leq x \leq 27$	$53 \leq x \leq 64$	$1 \leq x \leq 12$	$53 \leq x \leq 64$
$29 \leq y \leq 35$	$y = 8$	$29 \leq y \leq 35$	$y = 8$
(em $1 \leq x \leq 12$ é levemente superior)			

Os resultados obtidos com o oscilador em anel e o circuito 2 na placa do Cyclone II estão coerentes com os resultados escritos na literatura (Sedcole, 2006; Wong, 2007), onde pode-se verificar nos gráficos 3D uma forte correlação espacial na variação do atraso, visualizado como um plano construído com os valores médios. E, em cima desse plano, uma clara variação randômica do atraso.

Em relação ao circuito, 1 só foi encontrado um trabalho explicando o uso do circuito para testar atraso de propagação.

## **5.8 Trabalhos Futuros**

Para trabalhos futuros recomenda-se a criação de testes com capacidade de avaliar: elementos lógico; chaves de interconexão programáveis; e caminhos longos e curtos.

Outra sugestão é a automatização dos testes empregados nessa dissertação e o seu emprego em vários kits com o mesmo FPGA.

## 6 CONCLUSÕES

Uma vez realizados os seguintes procedimentos: ensaios de coleta de dados, processamento dos dados coletados e obtenção das informações geradas a partir dos dados coletados em conjunto com as respectivas análises, obtiveram-se resultados que permitem ao pesquisador apresentar o seguinte conjunto de conclusões:

Em relação à avaliação da metodologia adotada para identificar a existência de variabilidade do tempo de atraso de propagação de um circuito posicionado dentro de várias áreas de um FPGA, pôde-se concluir, de acordo com os resultados obtidos, que todos os recursos utilizados para analisar os valores medidos mostram a existência de variações *within-die*. Mas, o teste estatístico de hipótese para médias t-student mostra que as médias do circuito 2 e do oscilador em anel dos dois FPGA apresentam uma diferença significativa com um intervalo de confiança de 95% ( $P < 0,05$ ).

Os resultados obtidos na avaliação da metodologia permitem concluir que o circuito 1 necessita ter uma relação entre o “numero de ciclos” e à diferença de tempo entre o caminho mais lento e mais rápido para indicar falha no funcionamento do circuito devido ao atraso de propagação e que o circuito 2 não dá informações sobre o caminho mais rápido e a diferença de tempo entre os caminhos testados. Em relação aos outros circuito o funcionamento foi o esperado.

No que tange à identificação da variabilidade do tempo de atraso de propagação de um circuito posicionado dentro de várias áreas de um FPGA, objeto de estudo da presente dissertação, pôde-se identificar, de acordo com os resultados obtidos, a existência de variações *within-die*, independente do tipo de FPGA.

Os resultados obtidos na identificação da variabilidade do tempo de atraso permitem concluir que os circuitos digitais criados em um FPGA podem apresentar falhas no

funcionamento pois, tal como se indicou na literatura, a falha pode estar associada à variabilidade dos parâmetros elétricos. Portanto, recomenda-se que os circuitos implementados com FPGA, quando projetados para operar nas condições de desempenho próximo ao limite do FPGA, antes de serem usados, devessem ter o seu funcionamento testado para deixar uma boa margem de segurança.

Para isso, os circuitos adotados nessa dissertação podem ser úteis para avaliar outras aplicações em qualquer tipo de FPGA, comprovar a degradação de FPGA que sofreram radiações, verificar a frequência máxima de funcionamento de um circuito, identificar as falhas tanto do FPGA como do circuito, verificar a existência ou não de diferenças na propagação de um sinal por dois ou mais caminhos sobre teste iguais, além de servir como circuito comparador.

## REFERÊNCIAS

ABRAMOVICI, M.; STROUD, C. BIST-Based Delay-Fault Testing in FPGAs. In: IEEE INTERNATIONAL ON-LINE TESTING WORKSHOP (IOLTW'02), 8., 2002, Isle of Bendor, France. **Proceeding...** [S. l.]: IEEE, 2002. p. 1 – 4. ISBN: 0-7695-1641-6/02.

ALTERA CORPORATION. **DE2 Development and Education Board User Manual** version 1.4. San Jose, CA: Altera Corporation. 2006. 72 p. Disponível em: <ftp://ftp.altera.com/up/pub/Webdocs/>. Acesso em: 13 nov. 2009.

\_\_\_\_\_. **Cyclone II Device Handbook**, v. 1. San Jose, CA: Altera Corporation. 2008. 470 p. Disponível em: <http://www.altera.com/literature/lit-cyc2.jsp>. Acesso em: 28 fev. 2010.

\_\_\_\_\_. **DE2 Board**. [2009?]. Disponível em: <http://university.altera.com/materials/boards/de2/>. Acesso em: 17 fev. 2010).

BORKAR, S. at al. Parameter Variations and Impact on Circuits and Microarchitecture. In: ANNUAL ACM IEEE DESIGN AUTOMATION CONFERENCE, 40., 2003, Anaheim, CA, USA. **Proceedings ...** [S. l.]: IEEE, 2003. p. 338-342. ISBN:1-58113-688-9/03/0006

BROWN, S.; VRANESIC, Z. **Fundamentals of Digital Logic with VHDL Design**. 2.ed. New York: McGraw-Hill, 2005. 939. p. ISBN: 0-07-246085-7.

CARRO, L. **Projeto e Prototipação de Sistemas Digitais**. 1.ed. Porto Alegre: Editora da Universidade/UFRGS, 2001. 171 p. ISBN: 85-7025-589-6.

COSTA, C. **Projetos de Circuitos Digitais com FPGA**. 1.ed. São Paulo: Editora Érica Ltda., 2009. 206 p. ISBN: 978-85-365-0239-7.

D'AMORE, R. **VHDL Descrição e Síntese de Circuitos Digitais**. Rio de Janeiro: Editora LTC, 2005. 259. p. ISBN: 85-216-1452-7.

DAS, B. P. at al. Within-Die Gate Delay Variability Measurement Using Reconfigurable Ring Oscillator. **IEEE Transactions on Semiconductor Manufacturing**, [S. l.], v. 22, n. 2, p. 256-267, May 2009.

DIGILENT. **Spartan-3 Board**. [2009?]. Disponível em: <http://www.digilentinc.com/Products/Detail.cfm?Prod=S3BOARD>. Acesso em: 17 fev. 2010.

GHISSONI, S.; REIS, R. L. Otimização de Consumo de Potência e Atraso para Circuitos CMOS Utilizando SCCGs com Topologia Anti-radiação. In: WORKSHOP IBERCHIP, 15., 2009, Buenos Aires, Argentina. **Anais...** La Plata: Ediciones Cientificas Americanas, 2009. p. 334-337.

GIRARD, P. et al. An Efficient BIST Architecture for Delay Faults in the Logic Cells of Symmetrical SRAM-Based FPGAs. **Journal of Electronic Testing: theory and applications** [S. l.]: ©2006 Springer Science Business Media, LLC, n. 22, p. 161–172. 2006.

GONSALES, A. D. **Projeto de uma Nova Arquitetura de FPGA para Aplicações BIST e DSP**. 2002. 108 fls. Dissertação (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2002.

HANSELMAN, D.; LITTLEFIELD, B. **Matlab 6: curso completo**. 2ª Reimpressão. São Paulo: Pearson - Prentice Hall, 2007. p. 676. ISBN: 978-85-87918-56-7.

HARRIS, I. G.; MENON, P. R.; TESSIER, R. Bist-Based Delay Path Testing in FPGA Architectures. In: ITC INTERNATIONAL TEST CONFERENCE 2001 (ITC'01), 2001, Baltimore, Maryland. **Proceedings ...** [S. l.]: IEEE, 2001, p. 932-938. ISBN: 0-7803-7171-2.

KATSUKI, K. et al. A Yield and Speed Enhancement Scheme under Within-die Variations on 90nm LUT Array. In: IEEE 2005 CUSTOM INTEGRATED CIRCUITS CONFERENCE (CICC 2005), 27., 2005, San Jose, California. **Proceeding...** [S. l.]: IEEE, 2005. p. 601-604. ISBN: 0-7803-9023-7/05.

KOBAYASHI, K. et al. A Yield and Speed Enhancement Technique Using Reconfigurable Devices Against Within-Die Variations on the Nanometer Regime. In: INTERNATIONAL CONFERENCE ON FIELD PROGRAMMABLE LOGIC AND APPLICATIONS (FPL'06), 16., 2006, Madrid. **Proceeding...** [S. l.]: IEEE, 2006. p. 1-4. ISBN: 1-4244-0 312-X/06.

KUFLUOGLU, H.; ALAM, M. A. A Computational Model of NBTI and Hot Carrier Injection Time-Exponents for MOSFET Reliability. **Journal of Computational Electronics**. Netherlands: ©2005 Springer Science Business Media, n. 3. p. 165-169. 2004.

LEE, S.; KIM, B.; LEE, K. A Novel High-Speed Ring Oscillator for Multiphase Clock Generation Using Negative Skewed Delay Scheme. **IEEE Journal of Solid-State Circuits**. [S. l.], v. 32, n. 2, p.289-291, Feb. 1997.

LIN, Y.; HUTTON, M.; HE, L. Placement and Timing for FPGAs Considering Variations. In: INTERNATIONAL CONFERENCE ON FIELD PROGRAMMABLE LOGIC AND APPLICATIONS (FPL'06), 2006, Madrid. **Proceeding...** [S. l.]: IEEE, 2006. p. 1-7. ISBN: 1-4244-0312-X.

MAHMOODI, H.; MUKHOPADHYAY, S.; ROY, K. Estimation of Delay Variations due to Random-Dopant Fluctuations in Nanoscale CMOS Circuits. **IEEE Journal of Solid-State Circuits**. [S. l.], v. 40, n. 9, p.1787-1796, Sept. 2005.

MATSUMOTO, Y. et al. Performance and Yield Enhancement of FPGAs with Within-die Variation using Multiple Configurations. In: ACM/SIGDA INTERNATIONAL SYMPOSIUM ON FIELD-PROGRAMMABLE GATE ARRAY, 15., 2007. Monterey, California, USA. **Proceeding...** [S. l.]: ACM, 2007. p. 169–177.

MENON, P. R. et al. Design-Specific Path Delay Testing in Lookup-Table-Based FPGAs. In: **IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems**. New York, v. 25, n. 5, p. 867-877, May. 2006.

MUKHERJEE, A.; SKADRON, K. **Measuring Parameter Variation on an FPGA Using Ring Oscillators**. Jun. 2006. p. 6. Disponível em: <[http://www.cs.virginia.edu/~skadron/Papers/pv\\_tr2006\\_16.pdf](http://www.cs.virginia.edu/~skadron/Papers/pv_tr2006_16.pdf)>. Acesso em: 14 abr. 2010.

NASSIF, S. Delay Variability: sources, impacts and trends. In: INTERNATIONAL SOLID-STATE CIRCUITS CONFERENCE. DIGEST OF TECHNICAL PAPERS. (ISSCC), 2000, San Francisco, CA, USA. **Proceedings ...** [S. l.]: IEEE, 2000. p. 368-369.

NEUBERGER, G. **Protecting Digital Circuits Against Hold Time Violations Due to Process Variations**. 2007. 108 fls. Tese (Doutorado em Microeletrônica) – Programa de Pós-Graduação em Microeletrônica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2007.

NEUBERGER, G.; WIRTH, G.; REIS, R. (2009). Fault and Variability Modeling in Integrated Circuits with a Case Study. In: REIS, R. (Org.). **Advanced Topics on VLSI Design**. 1.ed. Porto Alegre: Editora da Universidade/UFRGS, 2009. p. 133-155. ISBN: 9788588425095.

NIKOLIC, B.; PANG, L. Measurements and Analysis of Process Variability in 90nm CMOS. In: INTERNATIONAL CONFERENCE ON SOLID-STATE AND INTEGRATED CIRCUIT TECHNOLOGY, 8., 2006, Shanghai, China. **Proceedings ...** [S. l.]: IEEE, 2006. p. 505-508.

SEDCOLE, P.; CHEUNG, P. Y. K. **Within-die Delay Variability in 90nm FPGAs and Beyond**. In: IEEE INTERNATIONAL CONFERENCE ON FIELD PROGRAMMABLE TECHNOLOGY (FPT 2006), Bangkok. **Proceeding...** [S. l.]: IEEE, 2006. p. 97-104.

SEDRA, A. S.; SMITH, K. C. **Microeletrônica**. 4.ed. São Paulo: Pearson Education-Makron Books. 2005. p. 1270. ISBN: 85-346-1044-4.

SMITH, J.; XIA, T. High-Precision Delay Testing of Virtex-4 FPGA Designs. In: IEEE INTERNATIONAL MIDWEST SYMPOSIUM ON CIRCUITS AND SYSTEMS (MWSCAS 2007), 50., Montreal, Canada. **Proceeding...** [S. l.]: IEEE, 2007. p. 1360-1353.

\_\_\_\_\_. High-Resolution Delay Testing of Interconnect Paths in Field-Programmable Gate Arrays. **IEEE Transactions on Instrumentation and Measurement**. [S. l.], v. 58, n.1, p. 187-195, Jan. 2009.

SPIEGEL, M. R. **Estatística**. 3.ed. São Paulo: Makron Books, 1994. 643 p. ISBN: 8534601208.

TAHOORI, M. B.; MITRA, S. Application-Dependent Delay Testing of FPGAs. **IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems**. [S. l.], v. 26, n. 3, p. 553-563, Mar. 2007.

WONG, J. S. J.; SEDCOLE, P.; CHEUNG, P. Y. K. **Self-characterization of Combinatorial Circuit Delays in FPGAs**. In: INTERNATIONAL CONFERENCE ON FIELD-PROGRAMMABLE TECHNOLOGY (ICFPT 2007), Kitakyushu. Japan. **Proceeding...** [S. l.]: IEEE, 2007. p 17-23.

XILINX. **Spartan-3 Starter Kit Board User Guide**. UG130. versão 1.1. May 13, 2005. 64 p. Disponível em: < [www.fri.uni-lj.si/file/62585/s3board-rm.pdf](http://www.fri.uni-lj.si/file/62585/s3board-rm.pdf) >. Acesso em: 20 maio 2009.

\_\_\_\_\_. **Spartan-3 Generation FPGA User Guide: Extended Spartan-3A, Spartan-3E, and Spartan-3 FPGA Families**. UG331. versão 1.6. Dec. 3, 2009. 524 p. Disponível em: < [http://www.xilinx.com/support/documentation/spartan-3\\_user\\_guides.htm](http://www.xilinx.com/support/documentation/spartan-3_user_guides.htm) >. Acesso em: 6 abr. 2010.

\_\_\_\_\_. **Spartan-3 FPGA Family Data Sheet**. DS099. Dec. 4, 2009. 217 p. Disponível em: < <http://www.xilinx.com/support/documentation/spartan-3.htm> >. Acesso em: 10 jan. 2010.

## APÊNDICES

### APÊNDICE A - CÓDIGOS FONTE DOS CIRCUITOS IMPLEMENTADOS

Neste apêndice apresenta-se, em VHDL, os códigos fonte dos circuitos implementados na dissertação.

#### A.1 Código Fonte do Circuito 1 para o Spartan 3

Para que o circuito 1 seja montado em todas as posições o arquivo main.ucf deve ser configurado conforme a segunda e terceira coluna da tabela 4.1.

##### A.1.1 Arquivo main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY main IS
    PORT ( sinal_chave_clear1_main : IN STD_LOGIC;
          sinal_chave_set1_main   : IN STD_LOGIC;
          sinal_out_count_main    : OUT STD_LOGIC_VECTOR(4 DOWNTO 0));
END main;
ARCHITECTURE structure OF main IS
    SIGNAL fio_out_q_latch_nor_main: STD_LOGIC;
    SIGNAL fio_out1_circuitoteste_main: STD_LOGIC;
    SIGNAL fio_out2_circuitoteste_main: STD_LOGIC;
    SIGNAL fio_out_ne3in_circuito_comparador_main: STD_LOGIC;
    SIGNAL fio_out_count_main : STD_LOGIC_VECTOR(4 DOWNTO 0);
BEGIN
    latch_nor1: latch_nor PORT MAP(in_set_latch_nor => sinal_chave_set1_main,
                                   in_clear_latch_nor => sinal_chave_clear1_main,
                                   out_q_latch_nor => fio_out_q_latch_nor_main);

    circuitoteste1: circuito_teste
        PORT MAP( in_circuitoteste => fio_out_q_latch_nor_main,
                 out1_circuitoteste=> fio_out1_circuitoteste_main,
                 out2_circuitoteste=> fio_out2_circuitoteste_main);

    circuito_comparador1: circuito_comparador
        PORT MAP ( sinal_in1_circuito_comparador=>fio_out1_circuitoteste_main,
                 sinal_in2_circuito_comparador => fio_out2_circuitoteste_main,

```

```

        sinal_out_ne3in_circuito_comparador=>
            fio_out_ne3in_circuito_comparador_main);
contador_assincrono1: contador_assincrono_mod_32
    PORT MAP( clk_count => fio_out_ne3in_circuito_comparador_main,
            reset_count => sinal_chave_clear1_main,
            q_count => fio_out_count_main);
    sinal_out_count_main <= fio_out_count_main;
END structure;

```

### A.1.2 Arquivo main.ucf

```

NET "sinal_out_count_main<4>" LOC = "P13" ;
NET "sinal_out_count_main<3>" LOC = "N14" ;
NET "sinal_out_count_main<2>" LOC = "L12" ;
NET "sinal_out_count_main<1>" LOC = "P14" ;
NET "sinal_out_count_main<0>" LOC = "K12" ;
NET "sinal_chave_set1_main" LOC = "K13" ;
NET "sinal_chave_clear1_main" LOC = "K14" ;
AREA_GROUP "AG_CIRCUITO_TESTE" RANGE = SLICE_X0Y0:SLICE_X26Y8;
INST "circuitoteste1" AREA_GROUP = "AG_CIRCUITO_TESTE" ;

```

### A.1.3 Arquivo package\_main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
PACKAGE package_main IS
    COMPONENT circuito_comparador
        PORT(sinal_in1_circuito_comparador : IN STD_LOGIC;
            sinal_in2_circuito_comparador : IN STD_LOGIC;
            sinal_out_ne3in_circuito_comparador : BUFFER STD_LOGIC);
    END COMPONENT;
    COMPONENT circuito_teste PORT( in_circuitoteste : IN STD_LOGIC;
        out1_circuitoteste : OUT STD_LOGIC;
        out2_circuitoteste : OUT STD_LOGIC);
    END COMPONENT;
    COMPONENT contador_assincrono_mod_32
        PORT(clk_count : IN STD_LOGIC;
            reset_count : IN STD_LOGIC;
            q_count : OUT STD_LOGIC_VECTOR(4 DOWNTO 0));
    END COMPONENT;
    COMPONENT jk_borda_neg PORT (clk_jk : IN STD_LOGIC;
        reset_jk : IN STD_LOGIC;
        j_jk : IN STD_LOGIC;
        k_jk : IN STD_LOGIC;
        q_jk : OUT STD_LOGIC);
    END COMPONENT;
    COMPONENT latch_nor PORT ( in_set_latch_nor : IN STD_LOGIC;

```

```

        in_clear_latch_nor : IN STD_LOGIC;
        out_q_latch_nor : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_ne_3in PORT (in1_ne_3in : IN STD_LOGIC;
                             in2_ne_3in : IN STD_LOGIC;
                             out_ne_3in : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_inversora PORT (in_inversora : IN STD_LOGIC;
                                out_inversora : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_nand PORT ( in1_nand : IN STD_LOGIC;
                             in2_nand : IN STD_LOGIC;
                             out_nand : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_ou PORT (in1_ou : IN STD_LOGIC;
                          in2_ou : IN STD_LOGIC;
                          out_ou : OUT STD_LOGIC);

END COMPONENT;
END package_main;

```

#### A.1.4 Arquivo latch\_nor.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY latch_nor IS
    PORT ( in_set_latch_nor : IN STD_LOGIC;
           in_clear_latch_nor : IN STD_LOGIC;
           out_q_latch_nor : OUT STD_LOGIC);
END latch_nor;
ARCHITECTURE structure OF latch_nor IS
    SIGNAL fio_out_ou1_latch: STD_LOGIC;
    SIGNAL fio_out_ou2_latch: STD_LOGIC;
    SIGNAL fio_out_inversora1_latch: STD_LOGIC:= '0';
    SIGNAL fio_out_inversora2_latch: STD_LOGIC:= '1';
BEGIN
    porta_ou1_latch: porta_ou PORT MAP( in1_ou => fio_out_inversora2_latch ,
                                       in2_ou => in_set_latch_nor,
                                       out_ou => fio_out_ou1_latch);
    porta_ou2_latch: porta_ou PORT MAP (in1_ou => fio_out_inversora1_latch,
                                       in2_ou => in_clear_latch_nor,
                                       out_ou => fio_out_ou2_latch);

    porta_inversora1_latch: porta_inversora
        PORT MAP( in_inversora => fio_out_ou1_latch,
                 out_inversora => fio_out_inversora1_latch);
    porta_inversora2_latch: porta_inversora
        PORT MAP ( in_inversora => fio_out_ou2_latch,
                 out_inversora => fio_out_inversora2_latch);

```

```

    out_q_latch_nor <= fio_out_inversora2_latch;
END structure;

```

### A.1.5 Arquivo circuito\_teste.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY circuito_teste IS
    GENERIC( n : INTEGER := 110);
    PORT(in_circuitoteste : IN STD_LOGIC;
        out1_circuitoteste : OUT STD_LOGIC;
        out2_circuitoteste : OUT STD_LOGIC);
END circuito_teste;
ARCHITECTURE structure OF circuito_teste IS
    SIGNAL fio_out_dasportas1_circuitoteste: STD_LOGIC_VECTOR( 0 TO n-1);
    SIGNAL fio_out_dasportas2_circuitoteste: STD_LOGIC_VECTOR( 0 TO n-1);
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_dasportas1_circuitoteste,
        fio_out_dasportas2_circuitoteste : SIGNAL IS TRUE;
BEGIN
    inversor1_circuitoteste: porta_inversora
        PORT MAP (in_inversora => in_circuitoteste,
            out_inversora => fio_out_dasportas1_circuitoteste(0));
    for_generate1: FOR i IN 1 to (n-1) GENERATE
        inversores_circuitoteste1: porta_inversora
            PORT MAP(in_inversora => fio_out_dasportas1_circuitoteste(i-1),
                out_inversora => fio_out_dasportas1_circuitoteste(i));
    END GENERATE for_generate1;
    inversor2_circuitoteste: porta_inversora
        PORT MAP (in_inversora => in_circuitoteste,
            out_inversora => fio_out_dasportas2_circuitoteste(0));
    for_generate2: FOR i IN 1 to (n-1) GENERATE
        inversores_circuitoteste2: porta_inversora
            PORT MAP ( in_inversora => fio_out_dasportas2_circuitoteste(i-1),
                out_inversora => fio_out_dasportas2_circuitoteste(i));
    END GENERATE for_generate2;
    out1_circuitoteste <= fio_out_dasportas1_circuitoteste(n-1);
    out2_circuitoteste <= fio_out_dasportas2_circuitoteste(n-1);
END structure;

```

### A.1.6 Arquivo circuito\_comparador.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;

```

```

USE work.package_main.all;
ENTITY circuito_comparador IS
    PORT(sinal_in1_circuito_comparador : IN STD_LOGIC;
          sinal_in2_circuito_comparador : IN STD_LOGIC;
          sinal_out_ne3in_circuito_comparador : BUFFER STD_LOGIC);
END circuito_comparador;
ARCHITECTURE structure OF circuito_comparador IS
    SIGNAL fio_out_ou_circuito_comparador : STD_LOGIC;
    SIGNAL fio_out_ne_circuito_comparador : STD_LOGIC;
    SIGNAL fio_out_ne3in_circuito_comparador : STD_LOGIC;
BEGIN
    porta_ou1: porta_ou PORT MAP(in1_ou => sinal_in1_circuito_comparador,
                                 in2_ou => sinal_in2_circuito_comparador,
                                 out_ou => fio_out_ou_circuito_comparador);
    porta_nand1: porta_nand PORT MAP(in1_nand => sinal_in1_circuito_comparador,
                                     in2_nand => sinal_in2_circuito_comparador,
                                     out_nand => fio_out_ne_circuito_comparador);

    porta_ne_3in1: porta_ne_3in
        PORT MAP (in1_ne_3in => fio_out_ou_circuito_comparador,
                  in2_ne_3in => fio_out_ne_circuito_comparador,
                  out_ne_3in => fio_out_ne3in_circuito_comparador);
    sinal_out_ne3in_circuito_comparador<= fio_out_ne3in_circuito_comparador;
END structure;

```

### A.1.7 Arquivo contador\_assincrono\_mod\_32.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY CONTADOR_ASSINCRONO_MOD_32 IS
    PORT( clk_count : IN STD_LOGIC;
          reset_count : IN STD_LOGIC;
          q_count : OUT STD_LOGIC_VECTOR(4 DOWNTO 0));
END CONTADOR_ASSINCRONO_MOD_32;
ARCHITECTURE behavioral OF CONTADOR_ASSINCRONO_MOD_32 IS
    SIGNAL high: STD_LOGIC;
    SIGNAL fio_q_count: std_logic_vector(4 DOWNTO 0);
BEGIN
    high <= '1'; -- conecta ao Vcc
    ff0: jk_borda_neg PORT MAP(j_jk => high, k_jk => high, clk_jk => clk_count,
                                reset_jk => reset_count, q_jk => fio_q_count(0));
    ff1: jk_borda_neg PORT MAP(j_jk => high, k_jk => high, clk_jk => fio_q_count(0),
                                reset_jk => reset_count, q_jk => fio_q_count(1));
    ff2: jk_borda_neg PORT MAP(j_jk => high, k_jk => high, clk_jk => fio_q_count(1),
                                reset_jk => reset_count, q_jk => fio_q_count(2));
    ff3: jk_borda_neg PORT MAP(j_jk => high, k_jk => high, clk_jk => fio_q_count(2),
                                reset_jk => reset_count, q_jk => fio_q_count(3));
    ff4: jk_borda_neg PORT MAP(j_jk => high, k_jk => high, clk_jk => fio_q_count(3),

```

```

reset_jk => reset_count, q_jk => fio_q_count(4));
    q_count <= fio_q_count;
END behavioral;

```

### A.1.8 Arquivo jk\_borda\_neg.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY jk_borda_neg IS
    PORT(clk_jk : IN STD_LOGIC;
         reset_jk : IN STD_LOGIC;
         j_jk : IN STD_LOGIC;
         k_jk : IN STD_LOGIC;
         q_jk : OUT STD_LOGIC);
END jk_borda_neg;
ARCHITECTURE behavioral OF jk_borda_neg IS
    SIGNAL q_temporario: STD_LOGIC;
BEGIN
    PROCESS (reset_jk, clk_jk, j_jk, k_jk)
    BEGIN
        IF reset_jk = '1' THEN
            q_temporario <= '0'; -- clear assincrono
        ELSIF (clk_jk'EVENT AND clk_jk = '0') THEN
            IF j_jk = '1' AND k_jk = '1' THEN
                q_temporario <= NOT q_temporario; -- COMUTACAO
            ELSIF j_jk = '1' AND k_jk = '0' THEN
                q_temporario <= '1'; -- setar
            ELSIF j_jk = '0' AND k_jk = '1' THEN
                q_temporario <= '0'; -- resetar
            END IF;
        END IF;
    END PROCESS;
    q_jk <= q_temporario; -- conecta o estado do flip-flop para a saida
END behavioral;

```

### A.1.9 Arquivo porta\_inversora.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_inversora IS
    PORT(in_inversora : IN std_logic;
         out_inversora : OUT std_logic);
END porta_inversora;
ARCHITECTURE behavioral OF porta_inversora IS
BEGIN

```

```

    out_inversora <= NOT(in_inversora);
END behavioral;

```

#### **A.1.10 Arquivo porta\_nand.vhd**

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_nand IS
    PORT(in1_nand : IN STD_LOGIC;
         in2_nand : IN STD_LOGIC;
         out_nand : OUT STD_LOGIC);
END porta_nand;
ARCHITECTURE behavioral OF porta_nand IS
BEGIN
    out_nand <= in1_nand NAND in2_nand;
END behavioral;

```

#### **A.1.11 Arquivo porta\_ne\_3in.vhd**

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_ne_3in IS
    PORT(in1_ne_3in : IN STD_LOGIC;
         in2_ne_3in : IN STD_LOGIC;
         out_ne_3in : BUFFER STD_LOGIC);
END porta_ne_3in;
ARCHITECTURE behavioral OF porta_ne_3in IS
    SIGNAL temp_in: STD_LOGIC := '0';
    SIGNAL temp_out: STD_LOGIC := '1';
BEGIN
    temp_out <= NOT(in1_ne_3in AND in2_ne_3in AND temp_in);
    out_ne_3in <= temp_out;
    temp_in <= temp_out;
END behavioral;

```

#### **A.1.12 Arquivo porta\_ou.vhd**

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_ou IS
    PORT(in1_ou : IN std_logic;
         in2_ou : IN std_logic;

```

```

        out_ou : OUT std_logic);
END porta_ou;
ARCHITECTURE Behavioral OF porta_ou IS
BEGIN
    out_ou <= (in1_ou OR in2_ou);
END Behavioral;

```

## A.2 Código Fonte do Circuito 1 para o Cyclone II

Neste apêndice só será apresentado os arquivos diferentes do circuito 1 do Spartan 3 ou as mudanças necessárias na configuração do Quartus.

### A.2.1 Arquivo main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;

ENTITY main IS
    PORT(sinal_chave_clear1_main : IN STD_LOGIC;
         sinal_chave_set1_main   : IN STD_LOGIC;
         sinal_out_count_main    : OUT STD_LOGIC_VECTOR(5 DOWNTO 0));
END main;
ARCHITECTURE structure OF main IS
    SIGNAL fio_out_q_latch_nor_main: STD_LOGIC;
    SIGNAL fio_out1_circuitoteste_main: STD_LOGIC;
    SIGNAL fio_out2_circuitoteste_main: STD_LOGIC;
    SIGNAL fio_out_ne3in_circuito_comparador_main: STD_LOGIC;
    SIGNAL fio_out_count_main : STD_LOGIC_VECTOR(5 DOWNTO 0);
BEGIN
    latch_nor1: latch_nor PORT MAP(in_set_latch_nor => sinal_chave_set1_main,
                                   in_clear_latch_nor => sinal_chave_clear1_main,
                                   out_q_latch_nor => fio_out_q_latch_nor_main);

    circuitoteste1 : circuito_teste
        PORT MAP( in_circuitoteste => fio_out_q_latch_nor_main,
                 out1_circuitoteste=> fio_out1_circuitoteste_main,
                 out2_circuitoteste=> fio_out2_circuitoteste_main);

    circuito_comparador1: circuito_comparador
        PORT MAP( sinal_in1_circuito_comparador=>fio_out1_circuitoteste_main,
                 sinal_in2_circuito_comparador => fio_out2_circuitoteste_main,
                 sinal_out_ne3in_circuito_comparador =>
                     fio_out_ne3in_circuito_comparador_main);

    contador_assincrono1: contador_assincrono_mod_64
        PORT MAP( clk_count => fio_out_ne3in_circuito_comparador_main,

```

```

        reset_count => sinal_chave_clear1_main,
        q_count => fio_out_count_main);
    sinal_out_count_main <= fio_out_count_main;
END structure;

```

### A.2.2 Arquivo main.ucf

No lugar deste arquivo deve ser configurado a ferramenta Pin Planner conforme a tabela A.1 e a área do arquivo circuito\_teste.vhd conforme a segunda, terceira e quarta colunas da tabela 4.2 com a ferramenta Chip Planner.

**Tabela A.1 – Relação entre a pinagem do FPGA e os sinais de entrada e saída do circuito 1 no Cyclone II.**

SINAL	PINAGEM
SINAL_OUT_COUNT_MAIN<5>	PIN_AD23
SINAL_OUT_COUNT_MAIN<4>	PIN_AD22
SINAL_OUT_COUNT_MAIN<3>	PIN_AC22
SINAL_OUT_COUNT_MAIN<2>	PIN_AB21
SINAL_OUT_COUNT_MAIN<1>	PIN_AF23
SINAL_OUT_COUNT_MAIN<0>	PIN_AE23
SINAL_CHAVE_SET1_MAIN	PIN_V2
SINAL_CHAVE_CLEAR1_MAIN	PIN_V1

### A.2.3 Arquivo package\_main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
PACKAGE package_main IS
    COMPONENT circuito_comparador
        PORT(sinal_in1_circuito_comparador : IN STD_LOGIC;
            sinal_in2_circuito_comparador : IN STD_LOGIC;
            sinal_out_ne3in_circuito_comparador : BUFFER STD_LOGIC);
    END COMPONENT;
    COMPONENT circuito_teste PORT(in_circuitoteste : IN STD_LOGIC;
        out1_circuitoteste : OUT STD_LOGIC;
        out2_circuitoteste : OUT STD_LOGIC);
    END COMPONENT;
    COMPONENT contador_assincrono_mod_64
        PORT(clk_count : IN STD_LOGIC;
            reset_count : IN STD_LOGIC;
            q_count : OUT STD_LOGIC_VECTOR(5 DOWNT0 0));
    END COMPONENT;
    COMPONENT jk_borda_neg PORT ( clk_jk : IN STD_LOGIC;

```

```

        reset_jk : IN STD_LOGIC;
        j_jk     : IN STD_LOGIC;
        k_jk     : IN STD_LOGIC;
        q_jk     : OUT STD_LOGIC);

END COMPONENT;
COMPONENT latch_nor PORT (in_set_latch_nor : IN STD_LOGIC;
                        in_clear_latch_nor : IN STD_LOGIC;
                        out_q_latch_nor   : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_ne_3in PORT(in1_ne_3in : IN STD_LOGIC;
                          in2_ne_3in : IN STD_LOGIC;
                          out_ne_3in  : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_inversora PORT( in_inversora : IN STD_LOGIC;
                               out_inversora : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_nand PORT( in1_nand : IN STD_LOGIC;
                          in2_nand : IN STD_LOGIC;
                          out_nand  : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_ou PORT (in1_ou : IN STD_LOGIC;
                       in2_ou : IN STD_LOGIC;
                       out_ou  : OUT STD_LOGIC);

END COMPONENT;
END package_main;

```

#### A.2.4 Arquivo circuito\_teste.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY circuito_teste IS
    GENERIC( n : INTEGER := 1350);-- NumerodeInversores
    PORT( in_circuitoteste : IN STD_LOGIC;
          out1_circuitoteste : OUT STD_LOGIC;
          out2_circuitoteste : OUT STD_LOGIC);
END circuito_teste;
ARCHITECTURE structure OF circuito_teste IS
    SIGNAL fio_out_dasportas1_circuitoteste: STD_LOGIC_VECTOR( 0 TO n-1);
    SIGNAL fio_out_dasportas2_circuitoteste: STD_LOGIC_VECTOR( 0 TO n-1);
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_dasportas1_circuitoteste,
        fio_out_dasportas2_circuitoteste : SIGNAL IS TRUE;
BEGIN
    inversor1_circuitoteste: porta_inversora
        PORT MAP ( in_inversora => in_circuitoteste,
                  out_inversora => fio_out_dasportas1_circuitoteste(0));
    for_generate1: FOR i IN 1 to (n-1) GENERATE

```

```

    inversores_circuitoteste1: porta_inversora
        PORT MAP ( in_inversora => fio_out_dasportas1_circuitoteste(i-1),
                  out_inversora => fio_out_dasportas1_circuitoteste(i));
END GENERATE for_generate1;
inversor2_circuitoteste: porta_inversora
    PORT MAP ( in_inversora => in_circuitoteste,
              out_inversora => fio_out_dasportas2_circuitoteste(0));
for_generate2: FOR i IN 1 to (n-1) GENERATE
    inversores_circuitoteste2: porta_inversora
        PORT MAP ( in_inversora => fio_out_dasportas2_circuitoteste(i-1),
                  out_inversora => fio_out_dasportas2_circuitoteste(i));
END GENERATE for_generate2;
out1_circuitoteste <= fio_out_dasportas1_circuitoteste(n-1);
out2_circuitoteste <= fio_out_dasportas2_circuitoteste(n-1);
END structure;

```

### A.2.5 Arquivo contador\_assincrono\_mod\_64.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY CONTADOR_ASSINCRONO_MOD_64 IS
    PORT ( clk_count : IN STD_LOGIC;
          reset_count : IN STD_LOGIC;
          q_count : OUT STD_LOGIC_VECTOR(5 DOWNTO 0));
END CONTADOR_ASSINCRONO_MOD_64;
ARCHITECTURE behavioral OF CONTADOR_ASSINCRONO_MOD_64 IS
    SIGNAL high: STD_LOGIC;
    SIGNAL fio_q_count: std_logic_vector(5 DOWNTO 0);
BEGIN
    high <= '1'; -- conecta ao Vcc
    ff0: jk_borda_neg PORT MAP ( j_jk => high, k_jk => high, clk_jk => clk_count,
                                reset_jk => reset_count, q_jk => fio_q_count(0));
    ff1: jk_borda_neg PORT MAP ( j_jk => high, k_jk => high, clk_jk => fio_q_count(0),
                                reset_jk => reset_count, q_jk => fio_q_count(1));
    ff2: jk_borda_neg PORT MAP ( j_jk => high, k_jk => high, clk_jk => fio_q_count(1),
                                reset_jk => reset_count, q_jk => fio_q_count(2));
    ff3: jk_borda_neg PORT MAP ( j_jk => high, k_jk => high, clk_jk => fio_q_count(2),
                                reset_jk => reset_count, q_jk => fio_q_count(3));
    ff4: jk_borda_neg PORT MAP ( j_jk => high, k_jk => high, clk_jk => fio_q_count(3),
                                reset_jk => reset_count, q_jk => fio_q_count(4));
    ff5: jk_borda_neg PORT MAP ( j_jk => high, k_jk => high, clk_jk => fio_q_count(4),
                                reset_jk => reset_count, q_jk => fio_q_count(5));

    q_count <= fio_q_count;
END behavioral;

```

### A.3 Código Fonte do Circuito 1 Modificado para o Spartan 3

Para que o circuito 1 modificado seja montado em todas as posições o arquivo main.ucf deve ser configurado conforme a quarta e quinta coluna da tabela 4.1.

#### A.3.1 Arquivo main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY main IS
    PORT( sinal_chave_clear  : IN STD_LOGIC;
          sinal_chave_in_um  : IN STD_LOGIC;
          sinal_chave_in_zero : IN STD_LOGIC;
          sinal_out_penultimoFF: OUT STD_LOGIC;
          sinal_out_ultimoFF  : OUT STD_LOGIC);
END main;
ARCHITECTURE structure OF main IS
    SIGNAL high: STD_LOGIC;
    SIGNAL fio_q_count: STD_LOGIC_VECTOR(12 DOWNTO 0);
    SIGNAL fio_out_ne3in_main: STD_LOGIC;
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_ne3in_main, fio_q_count : SIGNAL IS TRUE;
BEGIN
    high <= '1';
    sinal_out_penultimoFF <= fio_q_count(11);
    sinal_out_ultimoFF <= fio_q_count(12);
    teste1: circuito_teste PORT MAP(in1_circuitoteste => sinal_chave_in_um,
                                     in2_circuitoteste => sinal_chave_in_zero,
                                     out_circuitoteste => fio_out_ne3in_main);

    ff1: jk_borda_neg
        PORT MAP(j_jk => high, k_jk => high, clk_jk => fio_out_ne3in_main,
                reset_jk => sinal_chave_clear, q_jk => fio_q_count(0));
    ff2: jk_borda_neg
        PORT MAP (j_jk => high, k_jk => high, clk_jk => fio_q_count(0),
                reset_jk => sinal_chave_clear, q_jk => fio_q_count(1));
    ff3: jk_borda_neg
        PORT MAP (j_jk => high, k_jk => high, clk_jk => fio_q_count(1),
                reset_jk => sinal_chave_clear, q_jk => fio_q_count(2));
    ff4: jk_borda_neg
        PORT MAP (j_jk => high, k_jk => high, clk_jk => fio_q_count(2),
                reset_jk => sinal_chave_clear, q_jk => fio_q_count(3));
    ff5: jk_borda_neg
        PORT MAP (j_jk => high, k_jk => high, clk_jk => fio_q_count(3),
                reset_jk => sinal_chave_clear, q_jk => fio_q_count(4));

```



```

        j_jk : IN STD_LOGIC;
        k_jk : IN STD_LOGIC;
        q_jk : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_ne_3in PORT(in1_ne_3in : IN STD_LOGIC;
                            in2_ne_3in : IN STD_LOGIC;
                            out_ne_3in : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_nand PORT( in1_nand : IN STD_LOGIC;
                           in2_nand : IN STD_LOGIC;
                           out_nand : OUT STD_LOGIC);

END COMPONENT;
COMPONENT porta_ou PORT ( in1_ou : IN STD_LOGIC;
                          in2_ou : IN STD_LOGIC;
                          out_ou : OUT STD_LOGIC);

END COMPONENT;
END package_main;

```

### A.3.4 Arquivo circuito\_teste.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY circuito_teste IS
    PORT(in1_circuitoteste : IN STD_LOGIC;
         in2_circuitoteste : IN STD_LOGIC;
         out_circuitoteste : OUT STD_LOGIC);
END circuito_teste;
ARCHITECTURE structure OF circuito_teste IS
    SIGNAL fio_out_ou_teste: STD_LOGIC;
    SIGNAL fio_out_ne2in_teste: STD_LOGIC;
    SIGNAL fio_out_ne3in_teste: STD_LOGIC;
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_ou_teste, fio_out_ne2in_teste : SIGNAL IS TRUE;
BEGIN
    out_circuitoteste <= fio_out_ne3in_teste;
    porta_ou1: porta_ou PORT MAP ( in1_ou => in1_circuitoteste,
                                  in2_ou => in2_circuitoteste,
                                  out_ou => fio_out_ou_teste);
    porta_nand1: porta_nand PORT MAP ( in1_nand => in1_circuitoteste,
                                       in2_nand => in2_circuitoteste,
                                       out_nand => fio_out_ne2in_teste);
    porta_ne_3in1: porta_ne_3in PORT MAP ( in1_ne_3in => fio_out_ou_teste,
                                           in2_ne_3in => fio_out_ne2in_teste,
                                           out_ne_3in => fio_out_ne3in_teste);
END structure;

```

### A.3.5 Arquivo jk\_borda\_neg.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY jk_borda_neg IS
    PORT(clk_jk : IN STD_LOGIC;
         reset_jk : IN STD_LOGIC;
         j_jk : IN STD_LOGIC;
         k_jk : IN STD_LOGIC;
         q_jk : OUT STD_LOGIC);
END jk_borda_neg;
ARCHITECTURE behavioral OF jk_borda_neg IS
    SIGNAL q_temporario: STD_LOGIC;
BEGIN
    PROCESS (reset_jk, clk_jk, j_jk, k_jk)
    BEGIN
        IF reset_jk = '1' THEN
            q_temporario <= '0'; -- clear assincrono
        ELSIF (clk_jk'EVENT AND clk_jk = '0') THEN
            IF j_jk = '1' AND k_jk = '1' THEN
                q_temporario <= NOT q_temporario; -- COMUTACAO
            ELSIF j_jk = '1' AND k_jk = '0' THEN
                q_temporario <= '1'; -- setar
            ELSIF j_jk = '0' AND k_jk = '1' THEN
                q_temporario <= '0'; -- resetar
            END IF;
        END IF;
    END PROCESS;
    q_jk <= q_temporario;
END behavioral;

```

### A.3.6 Arquivo porta\_nand.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_nand IS
    PORT ( in1_nand : IN STD_LOGIC;
          in2_nand : IN STD_LOGIC;
          out_nand : OUT STD_LOGIC);
END porta_nand;
ARCHITECTURE behavioral OF porta_nand IS
BEGIN
    out_nand <= in1_nand NAND in2_nand;
END behavioral;

```

### A.3.7 Arquivo porta\_ne\_3in.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_ne_3in IS
    PORT( in1_ne_3in : IN STD_LOGIC;
          in2_ne_3in : IN STD_LOGIC;
          out_ne_3in : BUFFER STD_LOGIC);
END porta_ne_3in;
ARCHITECTURE behavioral OF porta_ne_3in IS
    SIGNAL temp_in: STD_LOGIC := '0';
    SIGNAL temp_out: STD_LOGIC := '1';
BEGIN
    temp_out <= NOT(in1_ne_3in AND in2_ne_3in AND temp_in);
    out_ne_3in <= temp_out;
    temp_in <= temp_out;
END behavioral;

```

### A.3.8 Arquivo porta\_ou.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_ou IS
    PORT( in1_ou : IN std_logic;
          in2_ou : IN std_logic;
          out_ou : OUT std_logic);
END porta_ou;
ARCHITECTURE Behavioral OF porta_ou IS
BEGIN
    out_ou <= (in1_ou OR in2_ou);
END Behavioral;

```

## A.4 Código Fonte do Circuito 1 Modificado para o Cyclone II

### A.4.1 Arquivo main.ucf

No lugar deste arquivo deve ser configurado a ferramenta Pin Planner conforme a tabela A.2 e a área do arquivo\_teste.vhd conforme a segunda, terceira e quarta colunas da tabela 4.2 com a ferramenta Chip Planner.



```

        clear_ar => clear_main,
        clk_ar => fio_out_ff3_main,
        out_ar => fio_out_ar_main);
    out_main <= fio_out_ar_main;
END structure;

```

### A.5.2 Arquivo main.ucf

```

NET "clear_main" LOC = "F12" ;
NET "clk_main" LOC = "C9" ;
NET "out_main" LOC = "K12" ;
AREA_GROUP "AG_CIRCUITO_TESTE" RANGE = SLICE_X0Y0:SLICE_X26Y8 ;
INST "teste1_main" AREA_GROUP = "AG_CIRCUITO_TESTE" ;

```

### A.5.3 Arquivo package\_main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
PACKAGE package_main IS
    COMPONENT analizador_resposta PORT( in1_ar : IN STD_LOGIC;
                                        in2_ar : IN STD_LOGIC;
                                        clear_ar : IN STD_LOGIC;
                                        clk_ar : IN STD_LOGIC;
                                        out_ar : BUFFER STD_LOGIC);

    END COMPONENT;
    COMPONENT arquivo_teste PORT(in_path_arqteste : IN STD_LOGIC;
                                clear_path_arqteste : IN STD_LOGIC;
                                clk_path_arqteste : IN STD_LOGIC;
                                out1_path_arqteste : OUT STD_LOGIC;
                                out2_path_arqteste : OUT STD_LOGIC);

    END COMPONENT;
    COMPONENT flipflop_dbn PORT( d_ffdbn : IN std_logic;
                                reset_ffdbn : IN STD_LOGIC;
                                clk_ffdbn : IN STD_LOGIC;
                                q_ffdbn : OUT std_logic);

    END COMPONENT;
    COMPONENT flipflop_dbp PORT( d_ffdbp : IN std_logic;
                                reset_ffdbp : IN STD_LOGIC;
                                clk_ffdbp : IN STD_LOGIC;
                                q_ffdbp : OUT std_logic);

    END COMPONENT;
    COMPONENT gerador_sequencia PORT(clear_gs : IN STD_LOGIC;
                                    clk_gs : IN STD_LOGIC;
                                    out_ff1_gs : OUT STD_LOGIC;
                                    out_ff3_gs : OUT STD_LOGIC);

    END COMPONENT;
    COMPONENT porta_e PORT( in1_e : IN std_logic;

```



```

                                clk_ffdbn => clk_gs,
                                q_ffdbn => fio_out_ff3_gs);
    out_ff1_gs <= fio_out_ff1_gs;
    out_ff3_gs <= fio_out_ff3_gs;
END structure;

```

### A.5.5 Arquivo arquivo\_teste.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY arquivo_teste IS
    GENERIC( n : INTEGER := 110);
    PORT( in_path_arqteste : IN STD_LOGIC;
          clear_path_arqteste : IN STD_LOGIC;
          clk_path_arqteste : IN STD_LOGIC;
          out1_path_arqteste : OUT STD_LOGIC;
          out2_path_arqteste : OUT STD_LOGIC);
END arquivo_teste;
ARCHITECTURE Structure OF arquivo_teste IS
    SIGNAL fio_out_ffdbn1_arqteste: STD_LOGIC;
    SIGNAL fio_out_dasportas1_arqteste: STD_LOGIC_VECTOR( 0 TO n-1);
    SIGNAL fio_out_dasportas2_arqteste: STD_LOGIC_VECTOR( 0 TO n-1);
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_ffdbn1_arqteste, fio_out_dasportas1_arqteste,
        fio_out_dasportas2_arqteste : SIGNAL IS TRUE;
BEGIN
    flipflop_dbn1_arqteste: flipflop_dbn PORT MAP( d_ffdbn => in_path_arqteste,
                                                reset_ffdbn=> clear_path_arqteste,
                                                clk_ffdbn => clk_path_arqteste,
                                                q_ffdbn => fio_out_ffdbn1_arqteste);

    inversor1_arqteste: porta_inversora
        PORT MAP ( in_inversora => fio_out_ffdbn1_arqteste,
                  out_inversora => fio_out_dasportas1_arqteste(0));
    for_generate1: FOR i IN 1 to (n-1) GENERATE
        inversores_arqteste1: porta_inversora
            PORT MAP(in_inversora => fio_out_dasportas1_arqteste(i-1),
                    out_inversora => fio_out_dasportas1_arqteste(i));
    END GENERATE for_generate1;
    inversor2_arqteste: porta_inversora
        PORT MAP ( in_inversora => fio_out_ffdbn1_arqteste,
                  out_inversora => fio_out_dasportas2_arqteste(0));
    for_generate2: FOR i IN 1 to (n-1) GENERATE
        inversores_arqteste2: porta_inversora
            PORT MAP ( in_inversora => fio_out_dasportas2_arqteste(i-1),
                    out_inversora => fio_out_dasportas2_arqteste(i));
    END GENERATE for_generate2;
    out1_path_arqteste <= fio_out_dasportas1_arqteste(n-1);

```

```

    out2_path_arqteste <= fio_out_dasportas2_arqteste(n-1);
END structure;

```

### A.5.6 Arquivo analizador\_resposta.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY analizador_resposta IS
    PORT(in1_ar : IN STD_LOGIC;
         in2_ar : IN STD_LOGIC;
         clear_ar : IN STD_LOGIC;
         clk_ar : IN STD_LOGIC;
         out_ar : OUT STD_LOGIC);
END analizador_resposta;
ARCHITECTURE structure OF analizador_resposta IS
    SIGNAL fio_out_porta_e1_ar: STD_LOGIC;
    SIGNAL fio_out_ffdbp1_ar: STD_LOGIC;
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_porta_e1_ar, fio_out_ffdbp1_ar : SIGNAL IS TRUE;
BEGIN
    porta_e1_ar: porta_e PORT MAP ( in1_e => in1_ar,
                                   in2_e => in2_ar,
                                   out_e => fio_out_porta_e1_ar);
    flipflop_dbp1_ar: flipflop_dbp PORT MAP(d_ffdbp => fio_out_porta_e1_ar,
                                           reset_ffdbp => clear_ar,
                                           clk_ffdbp => clk_ar,
                                           q_ffdbp => fio_out_ffdbp1_ar);

    out_ar <= fio_out_ffdbp1_ar;
END structure;

```

### A.5.7 Arquivo flipflop\_dbn.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY flipflop_dbn IS
    PORT(d_ffdbn : IN STD_LOGIC;
         reset_ffdbn : IN STD_LOGIC;
         clk_ffdbn : IN STD_LOGIC;
         q_ffdbn : OUT STD_LOGIC);
END flipflop_dbn;
ARCHITECTURE behavioral OF flipflop_dbn IS
BEGIN
    PROCESS ( reset_ffdbn, clk_ffdbn )
    BEGIN

```

```

        IF reset_ffdbn = '1' THEN
            q_ffdbn <= '0';
        ELSIF FALLING_EDGE(clk_ffdbn) THEN
            q_ffdbn <= d_ffdbn;
        END IF;
    END PROCESS;
END behavioral;

```

### A.5.8 Arquivo flipflop\_dbp.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY flipflop_dbp IS
    PORT(d_ffdbp : IN std_logic;
         reset_ffdbp : IN STD_LOGIC;
         clk_ffdbp : IN STD_LOGIC;
         q_ffdbp : OUT std_logic);
END flipflop_dbp;
ARCHITECTURE behavioral OF flipflop_dbp IS
BEGIN
    PROCESS ( reset_ffdbp, clk_ffdbp )
    BEGIN
        IF reset_ffdbp = '1' THEN
            q_ffdbp <= '0';
        ELSIF RISING_EDGE(clk_ffdbp) THEN
            q_ffdbp <= d_ffdbp;
        END IF;
    END PROCESS;
END behavioral;

```

### A.5.9 Arquivo porta\_e.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_ARITH.all;
USE ieee.std_logic_UNSIGNED.all;
ENTITY porta_e IS
    PORT(in1_e : IN std_logic;
         in2_e : IN std_logic;
         out_e : OUT std_logic);
END porta_e;
ARCHITECTURE behavioral OF porta_e IS
BEGIN
    out_e <= in1_e AND in2_e;
END behavioral;

```

### A.5.10 Arquivo porta\_ou.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_ARITH.all;
USE ieee.std_logic_UNSIGNED.all;
ENTITY porta_ou IS
    PORT( in1_ou : IN std_logic;
          in2_ou : IN std_logic;
          out_ou : OUT std_logic);
END porta_ou;
ARCHITECTURE Behavioral OF porta_ou IS
BEGIN
    out_ou <= (in1_ou OR in2_ou);
END Behavioral;

```

### A.5.11 Arquivo porta\_inversora.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_inversora IS
    PORT(in_inversora : IN std_logic;
          out_inversora : OUT std_logic);
END porta_inversora;
ARCHITECTURE behavioral OF porta_inversora IS
BEGIN
    out_inversora <= NOT(in_inversora);
END behavioral;

```

## A.6 Código Fonte do Circuito 2 para o Cyclone II

### A.6.1 Arquivo main.ucf

No lugar deste arquivo deve ser configurado a ferramenta Pin Planner conforme a tabela A.3 e a área do arquivo\_teste.vhd conforme a segunda, terceira e quarta colunas tabela 4.2 com a ferramenta Chip Planner.

**Tabela A.3 – Relação entre a pinagem do FPGA e os sinais de entrada e saída do circuito 2 no Cyclone II.**

SINAL	PINAGEM
OUT_MAIN	PIN_AE23
CLEAR_MAIN	PIN_N25
CLK_MAIN	PIN_P26

### A.6.2 Arquivo arquivo\_teste.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY arquivo_teste IS
    GENERIC( n : INTEGER := 1350);
    PORT( in_path_arqteste : IN STD_LOGIC;
          clear_path_arqteste : IN STD_LOGIC;
          clk_path_arqteste : IN STD_LOGIC;
          out1_path_arqteste : OUT STD_LOGIC;
          out2_path_arqteste : OUT STD_LOGIC);
END arquivo_teste;
ARCHITECTURE Structure OF arquivo_teste IS
    SIGNAL fio_out_ffdbn1_arqteste: STD_LOGIC;
    SIGNAL fio_out_dasportas1_arqteste: STD_LOGIC_VECTOR( 0 TO n-1);
    SIGNAL fio_out_dasportas2_arqteste: STD_LOGIC_VECTOR( 0 TO n-1);
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_ffdbn1_arqteste, fio_out_dasportas1_arqteste,
        fio_out_dasportas2_arqteste : SIGNAL IS TRUE;
BEGIN
    flipflop_dbn1_arqteste: flipflop_dbn PORT MAP( d_ffdbn => in_path_arqteste,
                                                reset_ffdbn => clear_path_arqteste,
                                                clk_ffdbn => clk_path_arqteste,
                                                q_ffdbn => fio_out_ffdbn1_arqteste);

    inversor1_arqteste: porta_inversora
        PORT MAP( in_inversora => fio_out_ffdbn1_arqteste,
                 out_inversora => fio_out_dasportas1_arqteste(0));
    for_generate1: FOR i IN 1 to (n-1) GENERATE
        inversores_arqteste1: porta_inversora
            PORT MAP ( in_inversora => fio_out_dasportas1_arqteste(i-1),
                      out_inversora => fio_out_dasportas1_arqteste(i));
    END GENERATE for_generate1;
    inversor2_arqteste: porta_inversora
        PORT MAP ( in_inversora => fio_out_ffdbn1_arqteste,
                 out_inversora => fio_out_dasportas2_arqteste(0));
    for_generate2: FOR i IN 1 to (n-1) GENERATE
        inversores_arqteste2: porta_inversora
            PORT MAP ( in_inversora => fio_out_dasportas2_arqteste(i-1),
                      out_inversora => fio_out_dasportas2_arqteste(i));
    END GENERATE for_generate2;

```

```

END GENERATE for _generate2;
out1_path_arqteste <= fio_out_dasportas1_arqteste(n-1);
out2_path_arqteste <= fio_out_dasportas2_arqteste(n-1);
END structure;

```

## A.7 Código Fonte do Oscilador em Anel para o SPARTAN 3

Para que o oscilador em anel seja montado em todas as posições o arquivo main.ucf deve ser configurado conforme a segunda e terceira coluna da tabela 4.1.

### A.7.1 Arquivo main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY main IS
    PORT(sinal_enable_main: IN STD_LOGIC;
         sinal_out_main: OUT STD_LOGIC);
END main;
ARCHITECTURE structure OF main IS
    SIGNAL fio_out_inv: STD_LOGIC;
    SIGNAL fio_out_nand: STD_LOGIC;
BEGIN
    inversores_serie1 : inversores_serie PORT MAP( in_inversores_serie => fio_out_nand,
                                                  out_inversores_serie => fio_out_inv);
    porta_nand1: porta_nand PORT MAP( in1_nand => sinal_enable_main,
                                     in2_nand => fio_out_inv,
                                     out_nand => fio_out_nand);

    sinal_out_main <= fio_out_inv;
END structure;

```

### A.7.2 Arquivo main.ucf

```

NET "sinal_enable_main" LOC = "H14" ;
NET "sinal_out_main" LOC = "A4" ;
AREA_GROUP "AG_MAIN" RANGE = SLICE_X0Y0:SLICE_X26Y8 ;
INST "main" AREA_GROUP = "AG_MAIN" ;

```

### A.7.3 Arquivo package\_main.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;

```

```

USE ieee.std_logic_unsigned.all;
PACKAGE package_main IS
    COMPONENT inversores_serie PORT( in_inversores_serie : IN STD_LOGIC;
                                     out_inversores_serie : OUT STD_LOGIC);

    END COMPONENT;
    COMPONENT porta_inversora PORT( in_inversora : IN STD_LOGIC;
                                     out_inversora : OUT STD_LOGIC);

    END COMPONENT;
    COMPONENT porta_nand PORT (in1_nand : IN STD_LOGIC;
                               in2_nand : IN STD_LOGIC;
                               out_nand : OUT STD_LOGIC);

    END COMPONENT;
END package_main;

```

#### A.7.4 Arquivo inversores\_serie.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY inversores_serie IS
    GENERIC( n : INTEGER := 110); -- Numero de Inversores
    PORT( in_inversores_serie : IN STD_LOGIC;
          out_inversores_serie : OUT STD_LOGIC);
END inversores_serie;
ARCHITECTURE structure OF inversores_serie IS
    SIGNAL fio_out_dasportas: STD_LOGIC_VECTOR( 0 TO n-1);
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_dasportas: SIGNAL IS TRUE;
BEGIN
    inversor1_circuitoteste: porta_inversora
        PORT MAP ( in_inversora => in_inversores_serie,
                  out_inversora => fio_out_dasportas(0));
    for_generate1: FOR i IN 1 to (n-1) GENERATE
        inversores_circuitotestel1: porta_inversora
            PORT MAP ( in_inversora => fio_out_dasportas(i-1),
                      out_inversora => fio_out_dasportas(i));
    END GENERATE for_generate1;
    out_inversores_serie <= fio_out_dasportas(n-1);
END structure;

```

#### A.7.5 Arquivo porta\_inversora.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_inversora IS

```

```

    PORT( in_inversora : IN std_logic;
          out_inversora : OUT std_logic);
END porta_inversora;
ARCHITECTURE behavioral OF porta_inversora IS
BEGIN
    out_inversora <= NOT(in_inversora);
END behavioral;

```

### A.7.6 Arquivo porta\_nand.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY porta_nand IS
    PORT ( in1_nand : IN STD_LOGIC;
          in2_nand : IN STD_LOGIC;
          out_nand : OUT STD_LOGIC);
END porta_nand;
ARCHITECTURE behavioral OF porta_nand IS
BEGIN
    out_nand <= in1_nand NAND in2_nand;
END behavioral;

```

## A.8 Código Fonte do Oscilador em Anel para o Cyclone II

### A.8.1 Arquivo main.ucf

No lugar deste arquivo deve ser configurado a ferramenta Pin Planner conforme a tabela A.4 e a área do arquivo\_teste.vhd conforme a segunda, terceira e quarta colunas da tabela 4.2 com a ferramenta Chip Planner.

**Tabela A.4– Relação entre a pinagem do FPGA e os sinais de entrada e saída do oscilador em anel no Cyclone II.**

SINAL	PINAGEM
SINAL_OUT_MAIN	PIN_M22
SINAL_ENABLE_MAIN	PIN_N25

### A.8.2 Arquivo inversores\_serie.vhd

```

LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;

```

```

USE ieee.std_logic_unsigned.all;
USE work.package_main.all;
ENTITY inversores_serie IS
    GENERIC( n : INTEGER := 1350);
    PORT(in_inversores_serie : IN STD_LOGIC;
         out_inversores_serie : OUT STD_LOGIC);
END inversores_serie;
ARCHITECTURE structure OF inversores_serie IS
    SIGNAL fio_out_dasportas: STD_LOGIC_VECTOR( 0 TO n-1);
    ATTRIBUTE KEEP : BOOLEAN;
    ATTRIBUTE KEEP OF fio_out_dasportas: SIGNAL IS TRUE;
BEGIN
    inversor1_circuitoteste: porta_inversora
        PORT MAP ( in_inversora => in_inversores_serie,
                  out_inversora => fio_out_dasportas(0));
    for_generate1: FOR i IN 1 to (n-1) GENERATE
        inversores_circuitotestel: porta_inversora
            PORT MAP( in_inversora => fio_out_dasportas(i-1),
                     out_inversora => fio_out_dasportas(i));
        END GENERATE for_generate1;
    out_inversores_serie <= fio_out_dasportas(n-1);
END structure;

```

## APÊNDICE B – SPARTAN 3 E CYCLONE II

Neste apêndice mostra-se, de forma resumida, algumas características dos kits e dos FPGA. Informações mais completas podem ser encontradas nos manuais dos fabricantes listados nas referências.

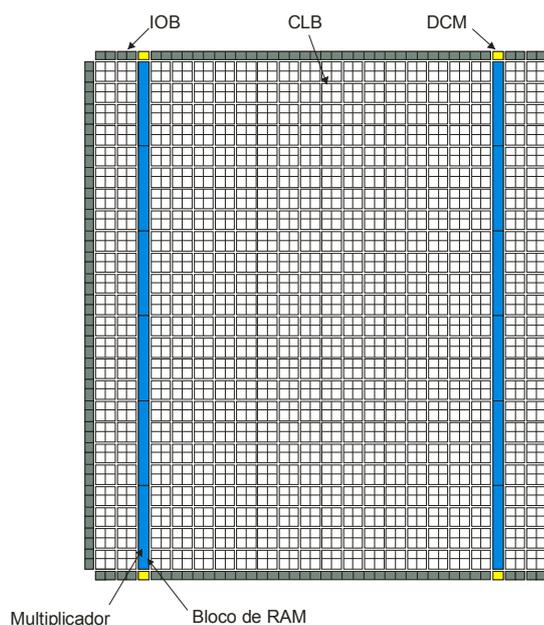
A placa Xilinx Spartan 3 Starter utiliza o FPGA XC3S200 com encapsulamento de 256 pinos e com uma taxa de transferência acima de 622Mb/s por I/O. Na tabela B.1 apresentam-se as características principais e na figura B.1 mostra-se a arquitetura.

**Tabela B.1 – Resumo das características do Spartan 3 (XILINX, 2009b).**

Portas do Sistema	Células Lógicas Equivalentes <sup>(1)</sup>	Matriz CLB (Uma CLB = 4 Slices)			
		Linhas	Colunas	Total CLBs	Total Slices
200K	4320	24	20	480	1920
Bits de RAM Distribuída	Bit de RAM por Bloco	Multiplicadores Dedicados	DCM	Máximo de I/O do Usuário	Máximo de Pares I/O Diferencial
30K	216K	12	4	173	76

Nota:

1. Célula Lógica = Look-Up Table (LUT) de 4 entradas mais um flip-flop tipo D.  
Células Lógicas Equivalentes = (Total de CLBs)x(8 células lógicas/CLB)x1,125.



**Figura B.1 - Arquitetura do Spartan 3 XC3S200FT256.**

A placa Altera DE2 utiliza o FPGA Cyclone II EP2C35 com encapsulamento de 672 pinos (ALTERA CORPORATION, 2009). Os dispositivos Cyclone II podem trabalhar com uma frequência de trabalho acima de 260MHz. Se for utilizado o sistema de sinalização diferencial de baixa tensão (LVDS - *Low-Voltage Differential Signaling*), os dispositivos Cyclone II podem receber dados com uma taxa máxima de 805 Mbps e transmitir dados com uma taxa máxima de 640 Mbps (ALTERA CORPORATION, 2008).

Na tabela B.2 apresentam-se as características principais e na figura B.2 mostra-se a arquitetura simplificada. A matriz lógica é formada de LAB que contém 16 elementos lógicos (LE). Um LE é a menor unidade de lógica. A LUT do LE é de quatro entradas. Com a ferramenta Chip Planner, a seqüência numérica do eixo X inclui os LAB, os multiplicadores e as RAM. Além disso, indica que na área de cor preta da figura 5.14 não existe LAB disponível para uso.

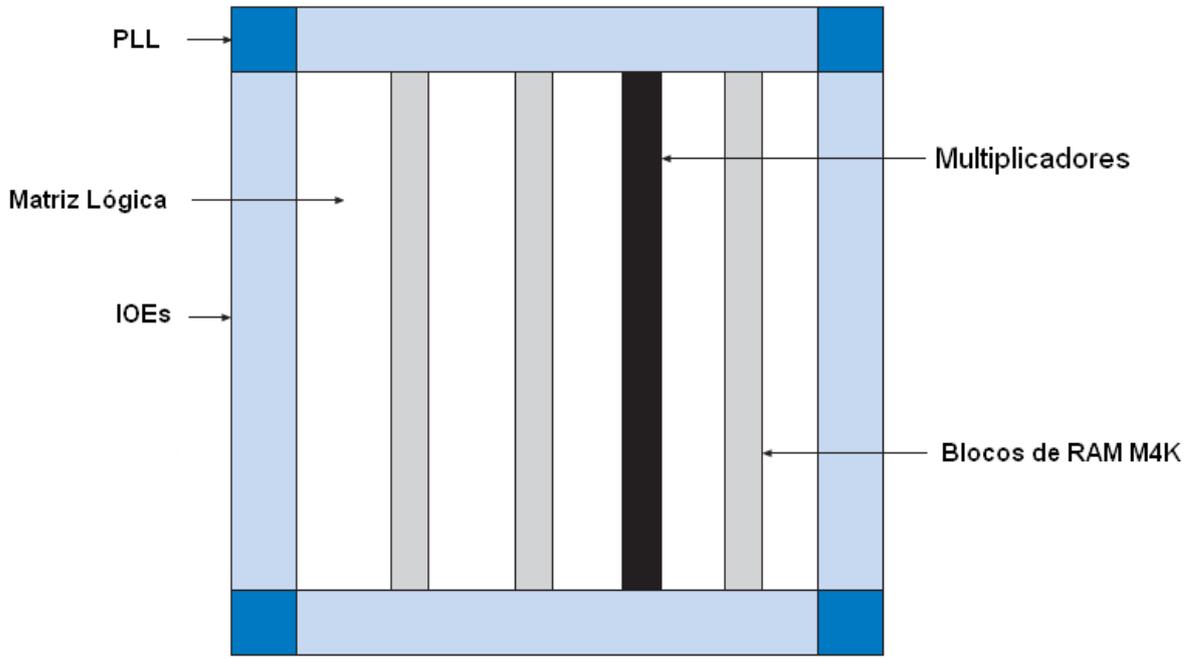
Os dois FPGA foram construídos com a mesma tecnologia de 90 nm (ALTERA CORPORATION, 2008; XILINX, 2009b) .

**Tabela B.2 – Resumo das características do Cyclone II (ALTERA CORPORATION, 2008).**

<b>LE</b>	<b>Blocos de RAM M4K (4 Kbits mais 512 bits de paridade)</b>	<b>Total de bits de RAM</b>	<b>Multiplicadores (18x18)</b>	<b>PLL<sup>(1)</sup></b>	<b>Número Máximo de Pinos de I/O do usuário</b>
32216	105	484840	35	4	475

Nota:

1. *Phase-Locked Loops.*



**Figura B.2 - Arquitetura do Cyclone II EP2C35F672.**

## APÊNDICE C - GRÁFICOS 3D COM OBSERVADOR POSICIONADO EM OUTRO LOCAL

Este apêndice contém outras figuras 3D com o observador posicionado em outro local com o objetivo de mostrar detalhes não observados na figura original.

### C.1 Gráficos 3D dos Circuitos Implementados no Spartan 3

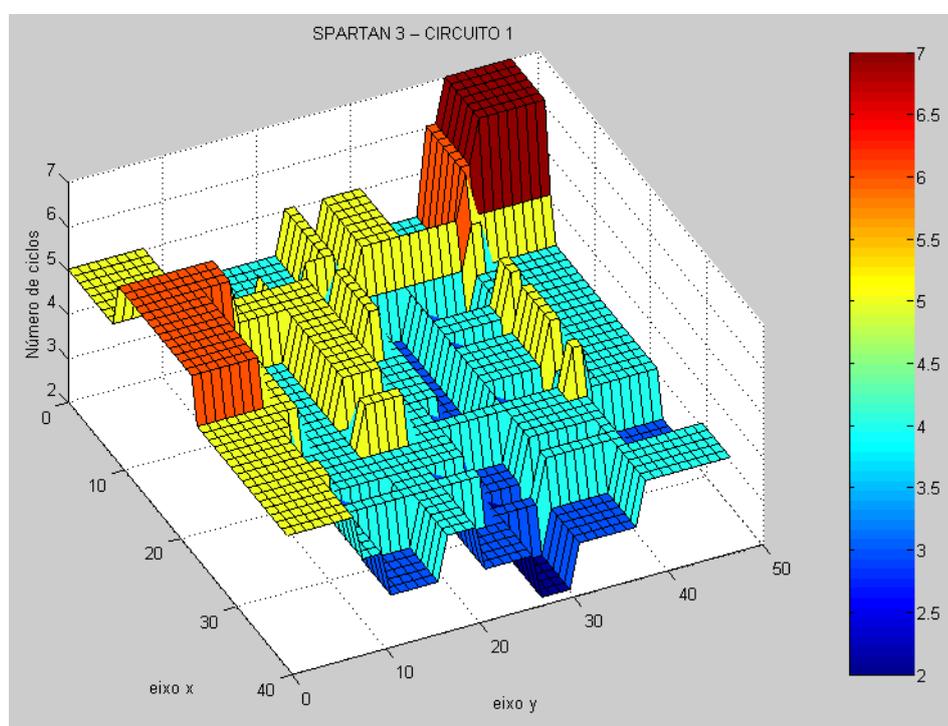
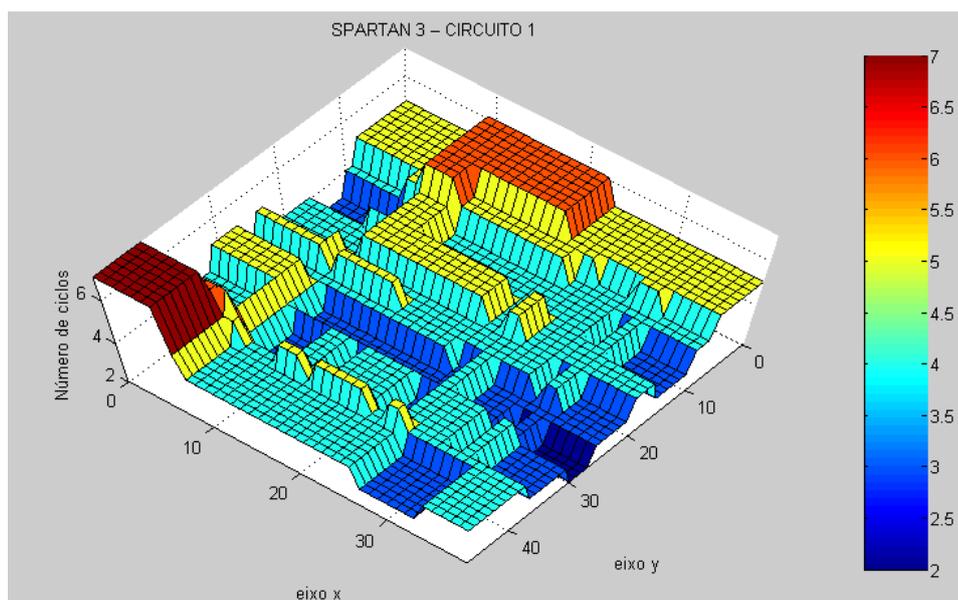
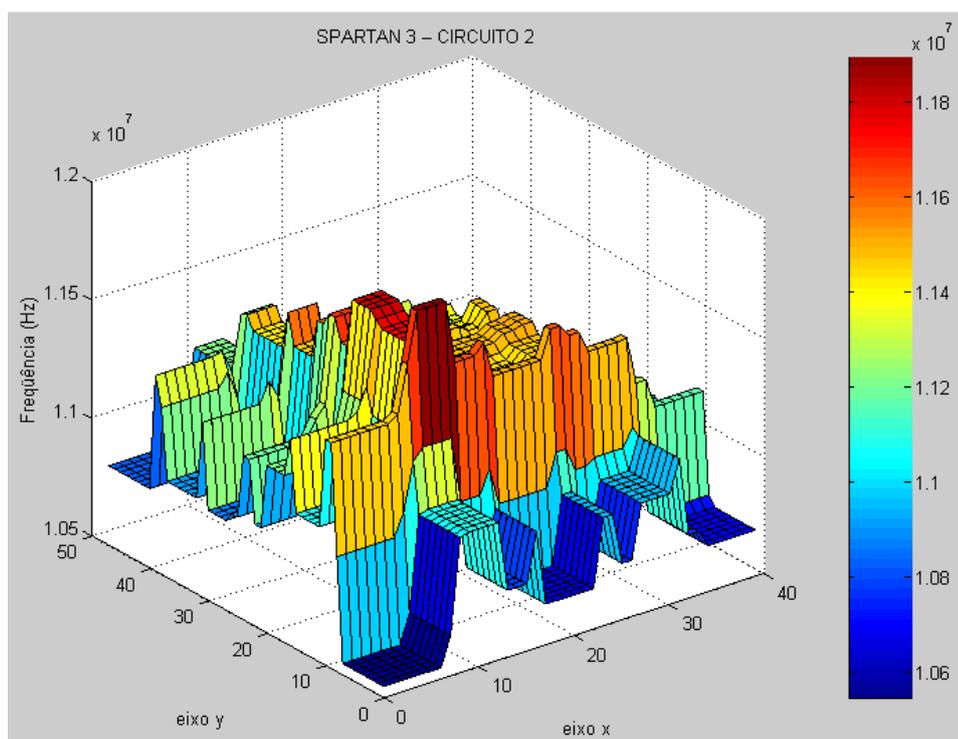


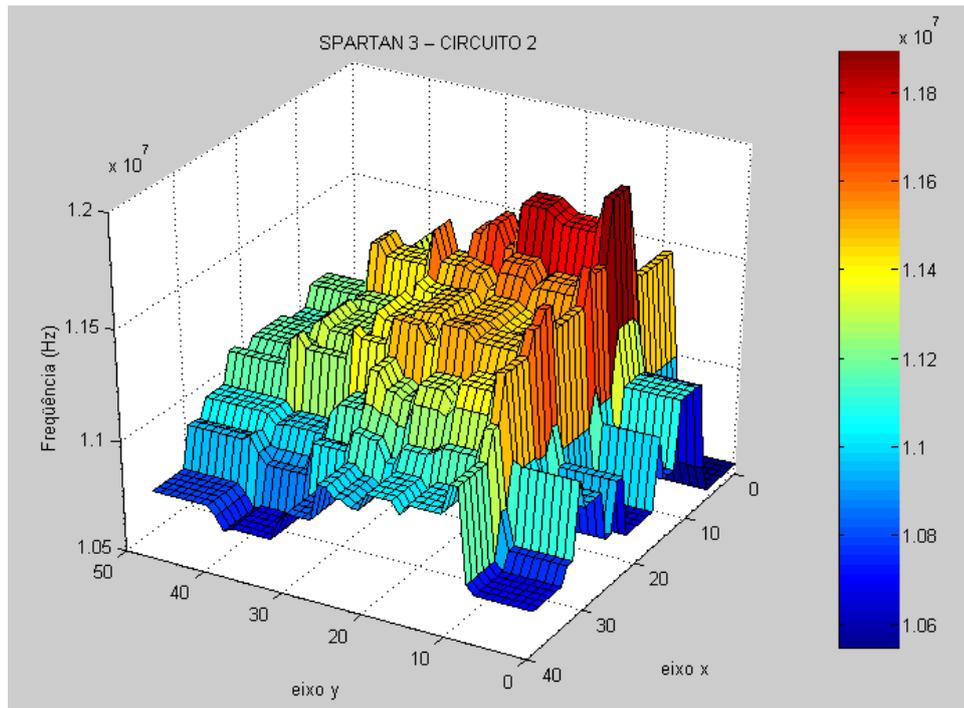
Figura C.1 – Gráfico opcional 1 da figura 5.5.



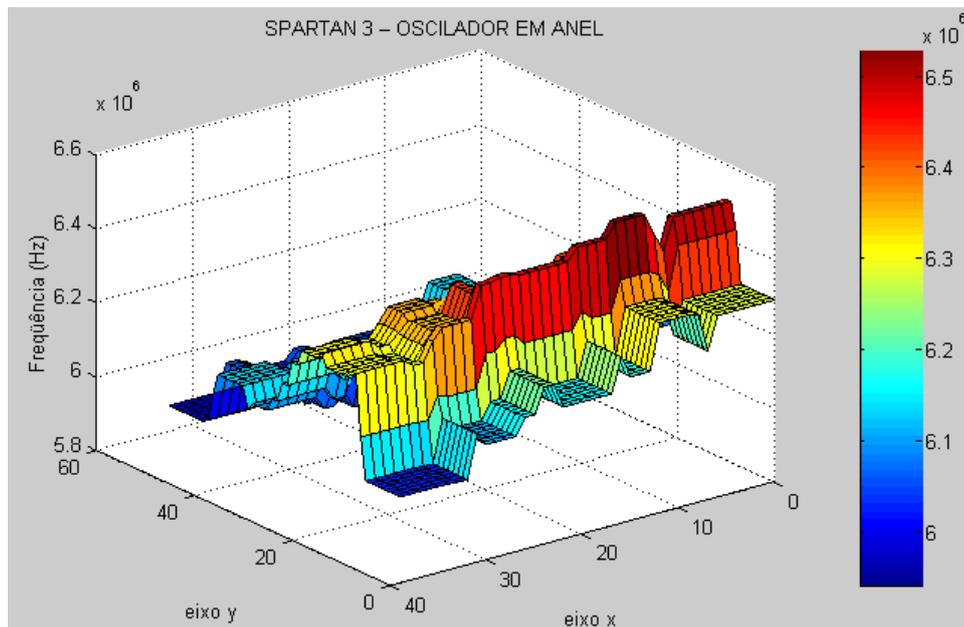
**Figura C.2 – Gráfico opcional 2 da figura 5.5.**



**Figura C.3 – Gráfico opcional 1 da figura 5.25.**



**Figura C.4 – Gráfico opcional 2 da figura 5.25.**



**Figura C.5 – Gráfico opcional 1 da figura 5.44.**

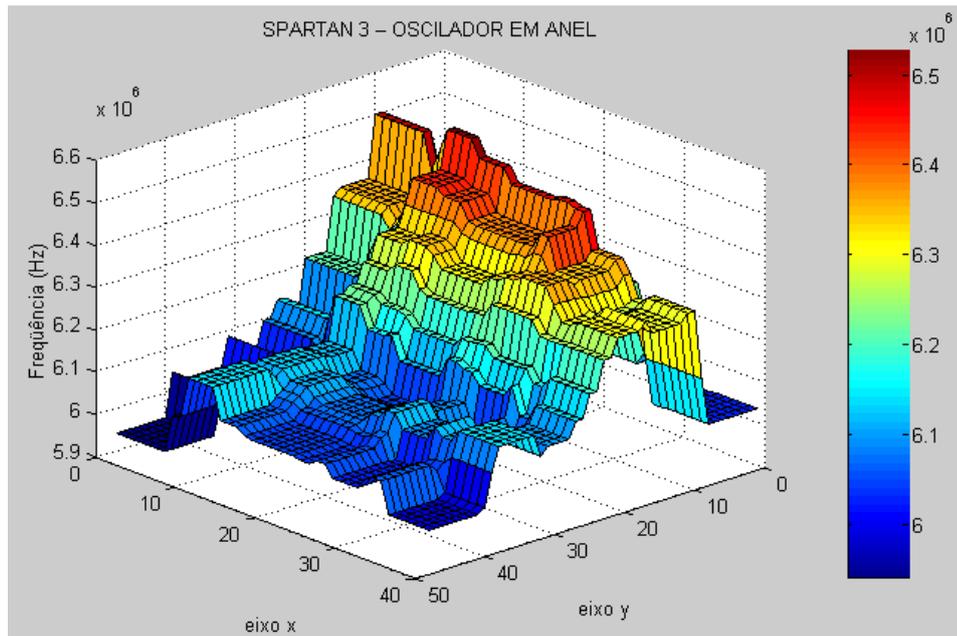


Figura C.6 – Gráfico opcional 2 da figura 5.44.

## C.2 Gráficos 3D dos Circuitos Implementados no Cyclone II

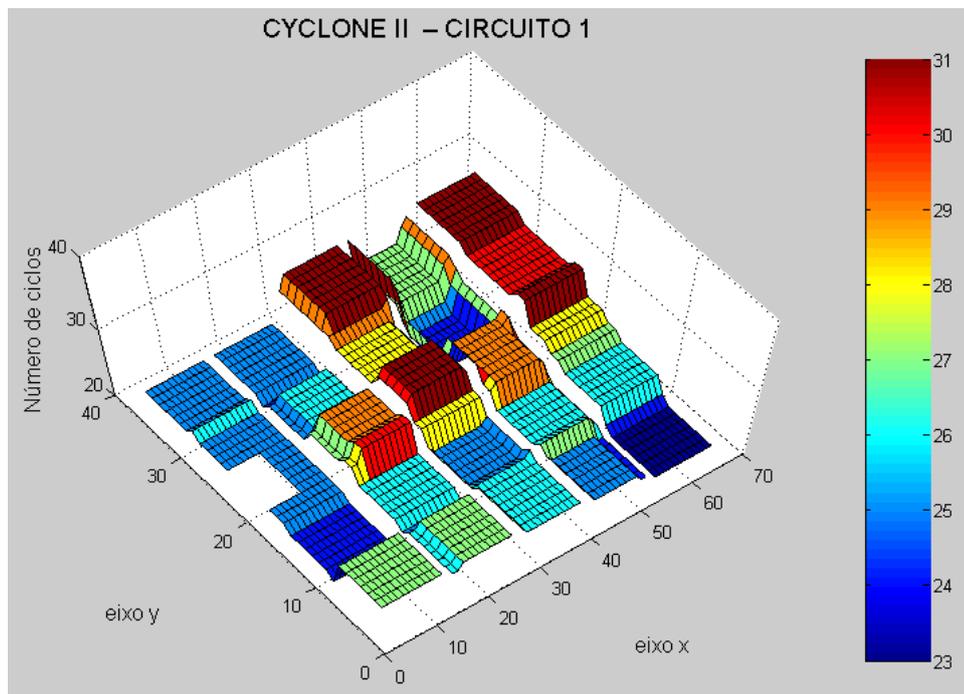
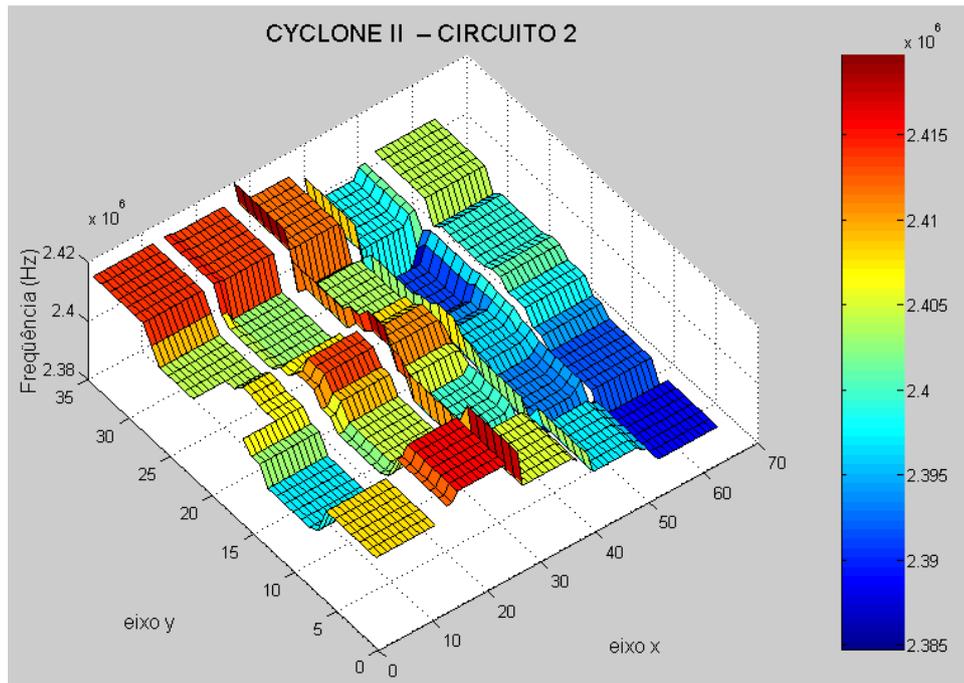
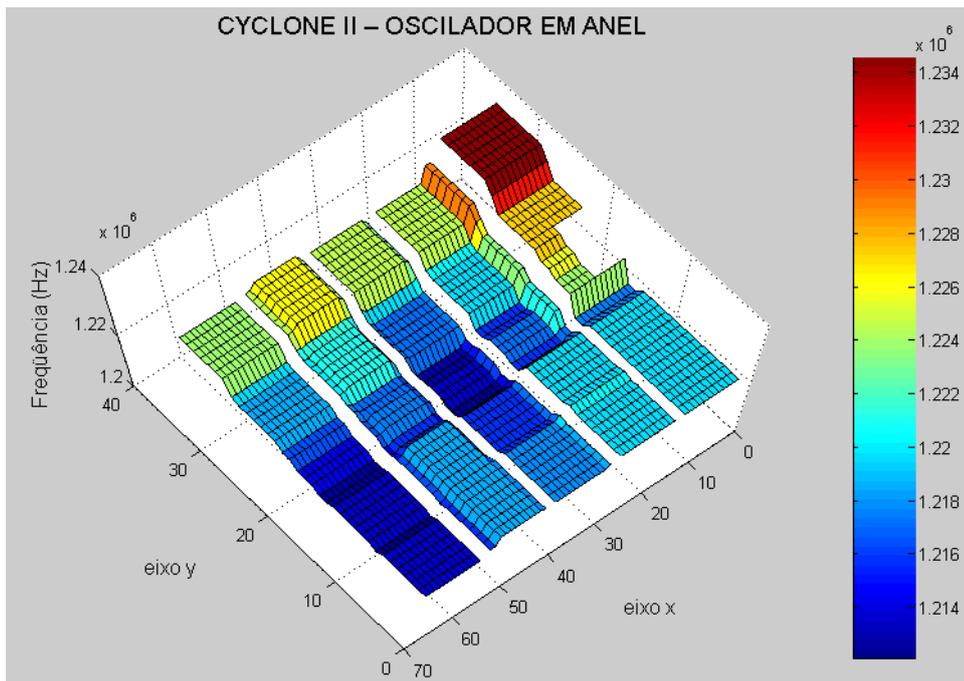


Figura C.7 – Gráfico opcional da figura 5.16.



**Figura C.8 – Gráfico opcional da figura 5.34.**



**Figura C.9 – Gráfico opcional da figura 5.52.**

## APÊNDICE D - ÁREAS DE RESTRIÇÃO

Neste apêndice mostra-se, em forma de figuras, as áreas de restrição adotadas durante a implementação dos circuitos.

### D.1 Áreas de Restrição do Spartan 3

Nas figuras abaixo apresentam-se as restrições usadas na tabela 4.1.

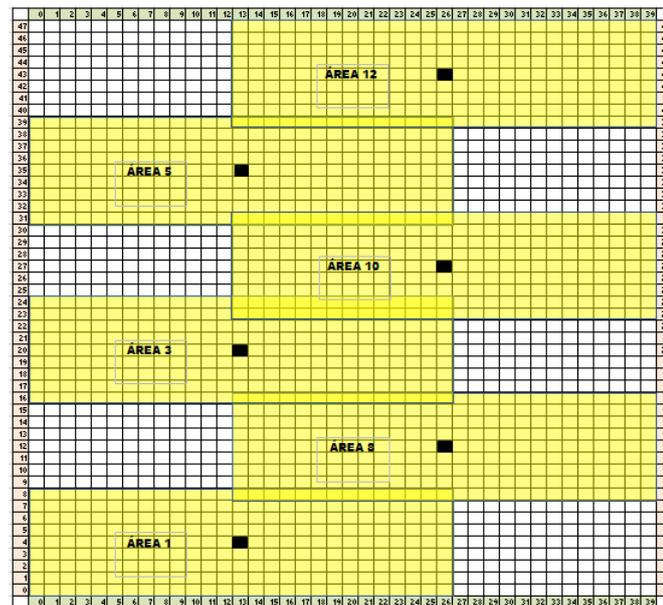


Figura D.1 – Áreas 1, 3, 5, 8, 10 e 12.

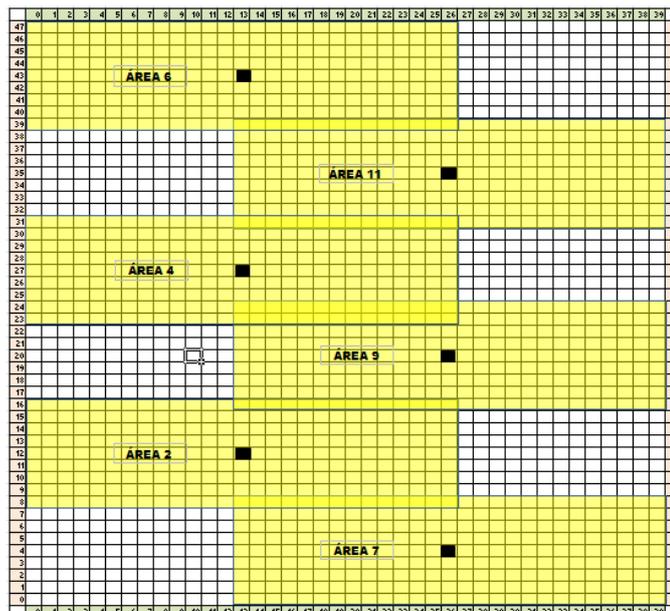


Figura D.2 – Áreas 2, 4, 6, 7, 9 e 11.

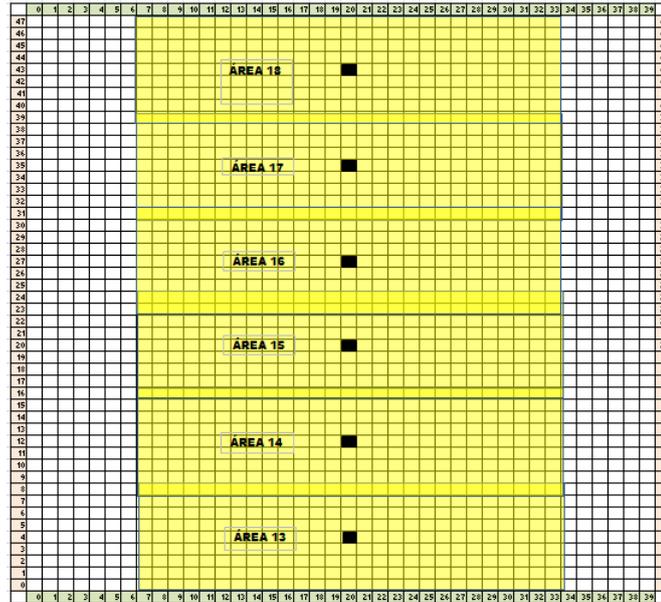


Figura D.3 – Áreas 13, 14, 15, 16, 17 e 18.

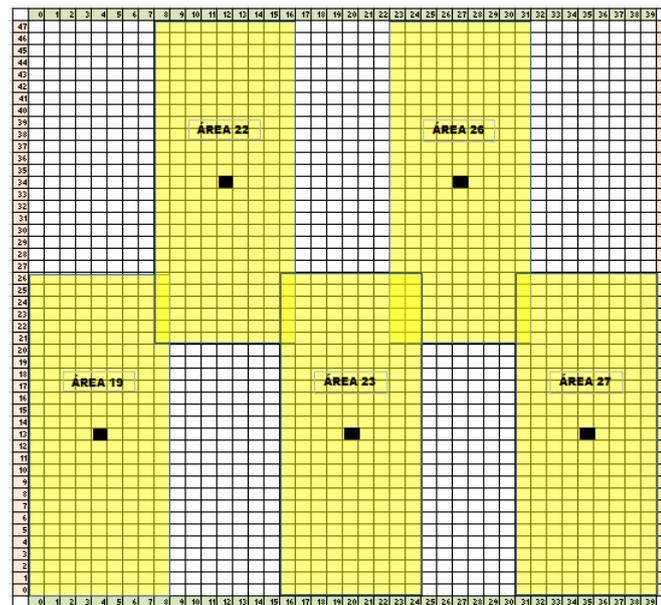


Figura D.4 – Áreas 19, 22, 23, 26 e 27.

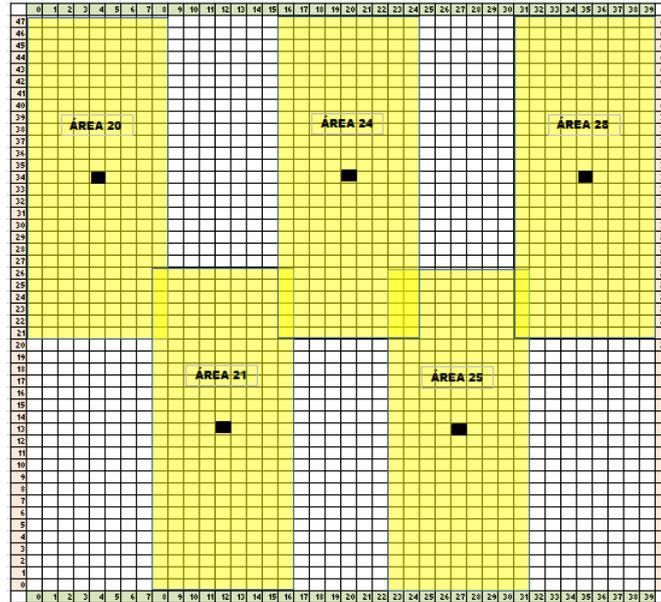


Figura D.5 – Áreas 20, 21, 24, 25 e 28.

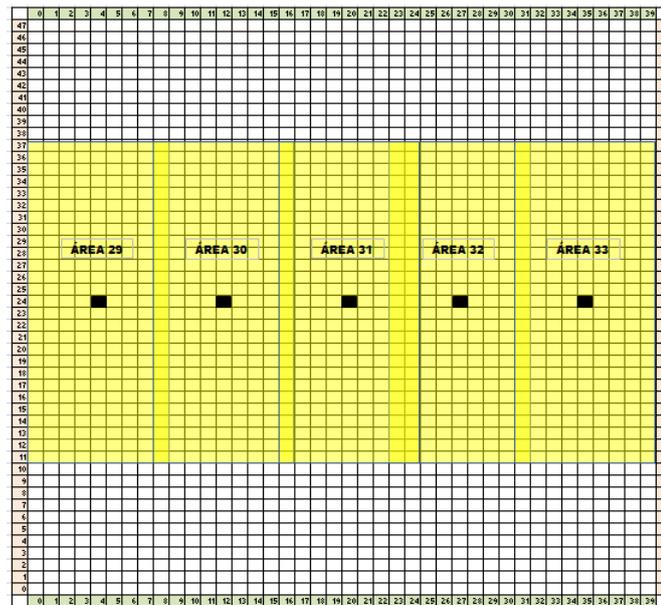


Figura D.6 – Áreas 29, 30, 31, 32 e 33.

## D.2 Áreas de Restrição do Cyclone II

Nas figuras abaixo apresentam-se as restrições usadas na tabela 4.2.

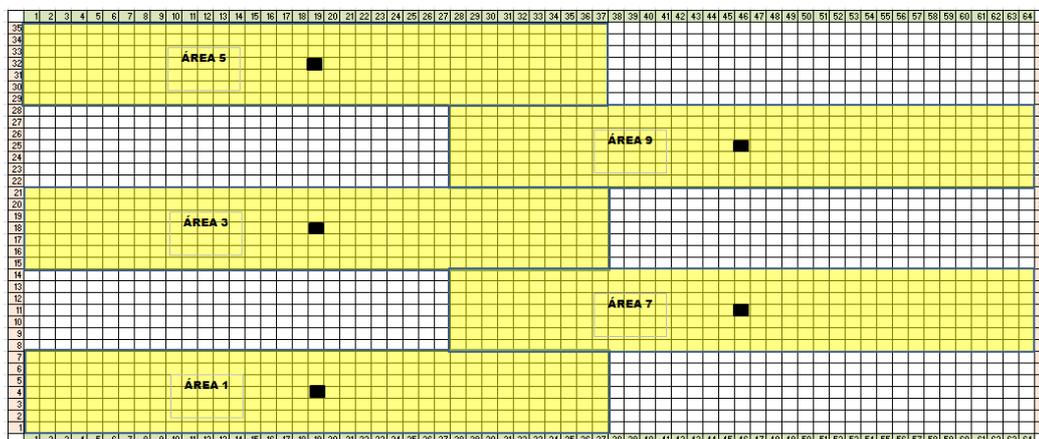


Figura D.7 – Áreas 1, 3, 5, 7 e 9.

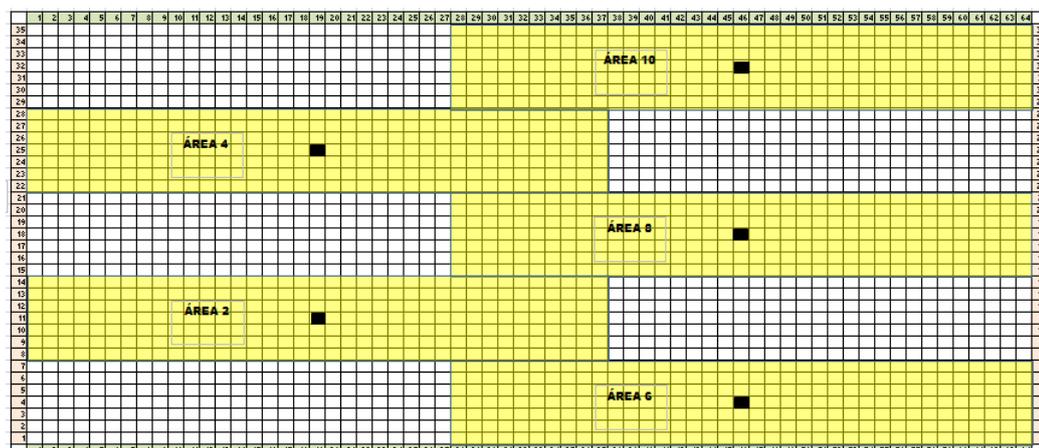


Figura D.8 – Áreas 2, 4, 6, 8 e 10.

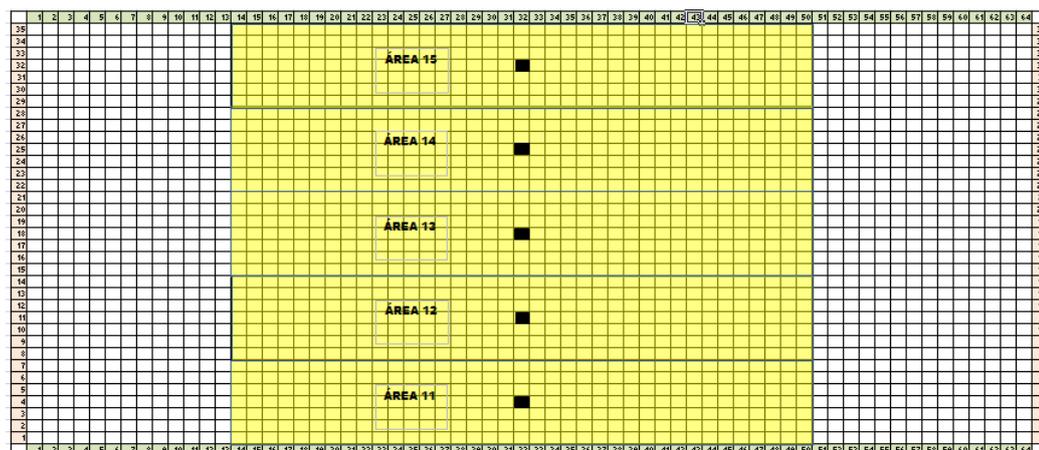


Figura D.9 – Áreas 11, 12, 13, 14 e 15.

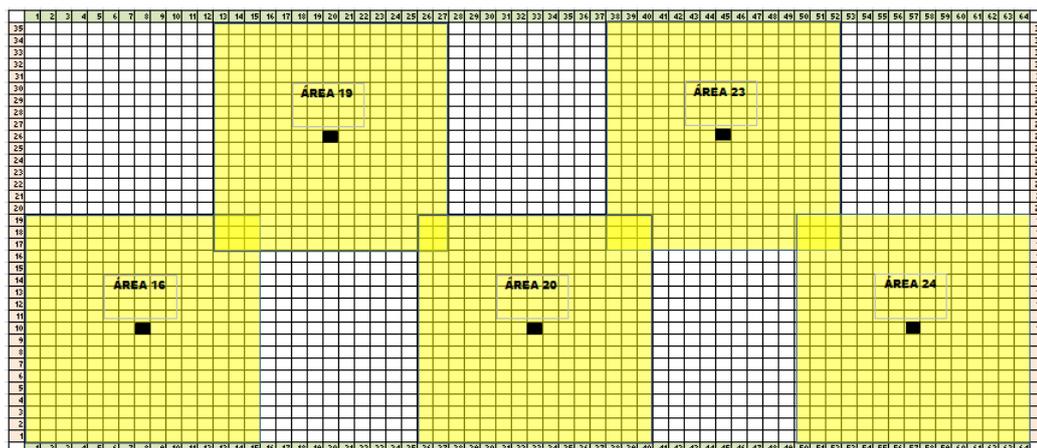


Figura D.10 – Áreas 16, 19, 20, 23 e 24.

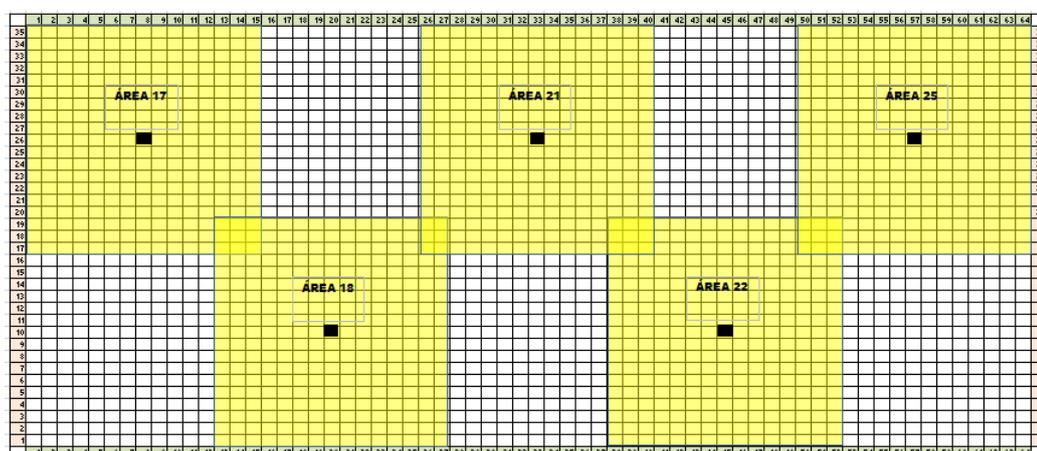


Figura D.11 – Áreas 17, 18, 21, 22 e 25.

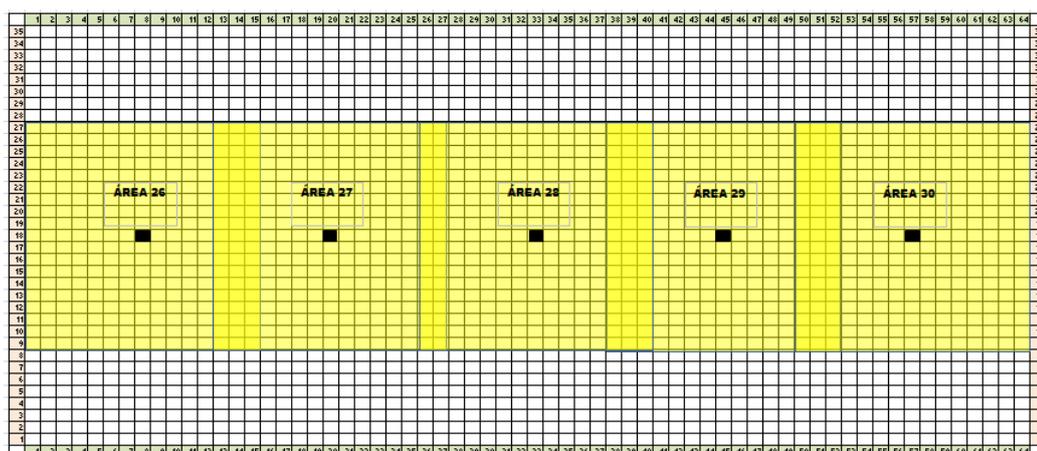


Figura D.12 – Áreas 26, 27, 28, 29 e 30.

## APÊNDICE E – FÓRMULAS ESTATÍSTICAS

Neste apêndice apresentam-se as fórmulas estatísticas utilizadas na dissertação.

### E.1 Média

A média de um conjunto de  $n$  números  $X_1, X_2, \dots, X_n$  é representada por  $\bar{X}$  e é definida por

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \quad (\text{E.1})$$

Onde:

$\bar{X}$  = média de todas as observações;

$X$  = valor de cada amostra;

$n$  = número de amostras;

### E.2 Desvio-Padrão

O desvio padrão de um conjunto de  $n$  números  $X_1, X_2, \dots, X_n$  é representada por  $\sigma$  e é definido por

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n-1}} \quad (\text{E.2})$$

Onde:

$\sigma$  = desvio padrão de todas as observações;

$\bar{X}$  = média de todas as observações;

$X$  = valor de cada amostra;

$n$  = número de amostras;

### E.3 Plano de Mínimos Quadrados

O plano de mínimos quadrados que se ajusta ao conjunto de pontos  $(X_1, Y_1, Z_1), (X_2, Y_2, Z_2), \dots, (X_n, Y_n, Z_n)$ , tem a equação:

$$Z = a_0 + a_1X + a_2Y \quad (\text{E.3})$$

em que as constantes  $a_0, a_1$  e  $a_2$  são determinadas mediante a resolução simultânea do sistema de equações:

$$\begin{cases} \Sigma Z = a_0N + a_1\Sigma X + a_2\Sigma Y \\ \Sigma XZ = a_0\Sigma X + a_1\Sigma X^2 + a_2\Sigma XY \\ \Sigma YZ = a_0\Sigma Y + a_1\Sigma XY + a_2\Sigma Y^2 \end{cases} \quad (\text{E.4})$$

### E.4 Média Móvel

Dado um conjunto de números:  $Z_1, Z_2, Z_3, \dots$  define-se uma média móvel de ordem  $n$ , a que é obtida pela seqüência das médias aritméticas

$$\frac{Z_1 + Z_2 + \dots + Z_n}{n}, \frac{Z_2 + Z_3 + \dots + Z_{n+1}}{n}, \frac{Z_3 + Z_4 + \dots + Z_{n+2}}{n}, \dots \quad (\text{E.5})$$