
PROTOTIPAGEM RÁPIDA DE MÓDULOS DE HARDWARE PARA UM AMBIENTE DE PROJETO CONJUNTO DE HW-SW

FAST PROTOTYPING OF HARDWARE MODULES FOR A HW-SW CODESIGN ENVIRONMENT

Frank Herman BEHRENS *

ABSTRACT

This work discusses the logic capacity specified for commercial Programmable Logic Devices (PLD's), from Altera and Xilinx, related to the gate array alternative. The gate count comparison is done based on some generic circuits and benchmark circuits. Preliminary results indicate a surprising gate count discrepancy for circuits designed in both PLDs and gate arrays. Gate array digital circuits of about 2500 equivalent gates can attain gate counts from 3113 to 5363 gates, roughly 25% to 115% more complexity, for Altera's development package. Considering Xilinx, for the same situation one can obtain estimated gate counts from 3672 to 7938, or 47% till 217% more gates. On the other hand, circuits coming from FPGAs' designs can reduce their gate counts by 60% when implemented on gate arrays. Therefore, one can not estimate a circuit complexity in PLD by simply computing *degree of utilization* versus *component logic capacity*. The transformation of a PLD into a gate array certainly has the benefit of a lower gate count. All these results can be applied to the development of a Hardware-Software Codesign environment, presently under study.

KEY WORDS: PLDs, benchmark, hardware design, integrated circuit

RESUMO

Este trabalho discute a capacidade lógica especificada para Dispositivos Lógicos Programáveis (PLD's), comerciais (EPLDs da Altera Corp. e FPGAs da Xilinx Inc.), em relação a alternativa *gate array*, comparando circuitos lógicos genéricos e circuitos específicos para *benchmark* em termos de sua complexidade, esta expressa em portas lógicas equivalentes. Os resultados preliminares indicam uma surpreendente discrepância entre a contagem de portas lógicas nas metodologias de projeto *gate array* e PLD, inclusive, não havendo uma proporção entre estes valores. Circuitos digitais de cerca de 2500 portas em *gate array* podem atingir, no ambiente Altera, contagens estimadas na faixa de 3113 a 5363 portas equivalentes, correspondendo a um aumento de 25% a 115%. No ambiente Xilinx, obteve-se contagens estimadas na faixa de 3672 a 7938, ou seja, 47% a 217%. Por outro lado, no sentido inverso de implementação, circuitos inicialmente projetados em FPGAs, quando passados para *gate array*, sofrem uma redução da contagem de portas lógicas de cerca de 60%. Portanto, não se pode estimar a complexidade de circuitos projetados em PLDs com base apenas no produto taxa de utilização x capacidade lógica do componente. A transformação de um PLD em *gate array* é certamente beneficiada pela menor contagem de portas lógicas resultante. Estes resultados podem ser utilizados na definição de alternativas de implementação de ASIC's (Application Specific Integrated Circuits), para um sistema de Projeto Conjunto de Hardware-Software, presentemente sob estudo.

PALAVRAS-CHAVE: PLDs, benchmark, projeto de hardware, circuito integrado

(*) Professor Titular do Instituto de Informática da Puccamp.

Este trabalho foi inicialmente conduzido no Instituto de Microeletrônica da Fundação Centro Tecnológico para Informática, FCTI, Campinas, São Paulo.

1. INTRODUÇÃO

A idéia de prototipagem rápida de circuitos integrados para processamento de dados vem sendo considerada uma alternativa viável para o desenvolvimento de hardware específico, segundo algumas aplicações de uso específico.

Sistemas computacionais são, atualmente, baseados em hardware padrão (computadores pessoais, servidores, estações de trabalho, etc), configurados segundo programas aplicativos (software) desenvolvidos de acordo com alguma aplicação em vista.

Hoje em dia, aparentemente não há condições econômicas para desenvolvimento de sistemas ou partes de sistemas de hardware, resultando na utilização em larga escala de sistemas padrão, que são mais econômicos e mais flexíveis do ponto de vista do usuário. Esta realidade é particularmente verdade na atual engenharia computacional brasileira neste final década. Não se vislumbra facilmente condições econômicas e técnicas de competitividade para sistemas de hardware totalmente nacionais.

Entretanto, surgiu recentemente a idéia de **algoritmos em hardware**, sugerindo que a implementação física de circuitos digitais, que realizem processamento específico, pode levar à otimização de sistemas, quanto a um desempenho e velocidade maiores do que a alternativa usual de implementação totalmente em software.

Esta metodologia de projeto de sistemas tem o nome de **Projeto Conjunto de Hardware e Software (HW/SW Codesign)** [1], [2] e visa sobretudo desenvolver um sistema integrado de especificação de um sistema computacional, que, em algum momento adequado do projeto, possa decidir qual parte do algoritmo será codificada em software e qual parte tem vantagens em ser implementada em hardware [3].

Normalmente, esta implementação toma a forma de um circuito ou conjunto de circuitos integrados de aplicação específica (ASICs - Application Specific Integrated Circuits), que serão eletricamente conectados a algum sistema de hardware padrão, que rodará conjuntamente a parte de software. A parte de hardware funciona, neste caso, analogamente a um co-processador do sistema padrão, acelerando a realização das funções relativas à parte do algoritmo codificada em hardware.

A seguir, discute-se algumas alternativas de implementação de ASICs dentro da idéia de prototipagem rápida.

2. METODOLOGIAS PARA ASICs

Até o início dos anos 80, as metodologias *gate array* e *standard cell* constituíam as formas de implementação de ASICs mais usuais, úteis para prototipagem de CIs ou mesmo para produção em quantidades até 1.000 a 20.000 peças por ano.

Com o surgimento da segunda geração de PLDs, no final dos anos 80, e a sua rápida evolução em capacidade lógica e desempenho, criou-se dúvidas para o usuário quanto a escolha da melhor alternativa para implementação de ASICs, principalmente pela diversidade de arquiteturas e elementos de programação (fusíveis, anti-fusíveis, células SRAM, EPROM e E2PROM) [4].

Os *gate arrays* atuais atingem capacidades lógicas na faixa de 100K portas equivalentes, existindo, entretanto, famílias que cobrem a faixa de 2K a 20K portas de maneira eficiente em termos de desempenho e custo [5]. Os PLDs mais recentes, segundo os fabricantes, possuem de 10K a 20K portas equivalentes [5] [6]. Porém, a maneira como é estimada sua capacidade lógica é questionável, devido à manipulação lógica de síntese e mapeamento realizada pelos programas de desenvolvimento.

Este trabalho apresenta uma análise comparativa entre uma matriz *gate array* [8] e os componentes PLDs, particularmente os EPLDs da Altera e FPGAs da Xilinx, segundo a sua capacidade de implementação de circuitos digitais. Não se objetiva aqui comparar as famílias de PLDs destes fabricantes entre si, por ser este um assunto já abordado por estudos anteriores, tais como o PREP [9] [10]. Cabe ressaltar que há características distintas entre eles: o ambiente da Xilinx permite a intervenção manual no roteamento de interconexão entre componentes lógicos, o que não ocorre com o da Altera; por outro lado, o ambiente da Altera possibilita o particionamento automático do circuito em vários componentes quando ele não couber em apenas um PLD, o que não é viável com o da Xilinx.

Portanto, este trabalho avalia, sob um ponto de vista prático, o limite de capacidade lógica dos componentes PLDs em relação à alternativa de implementação em *gate array*, sendo que esta última permite uma avaliação mais precisa da complexidade dos circuitos digitais usados como padrão de comparação.

3. A CAPACIDADE LÓGICA DE PLDs

A medida da complexidade de circuitos digitais, expressa em portas equivalentes, tornou-se muito utilizada devido à possibilidade de se estimar o tamanho de placas de circuito impresso (PCB), de pastilhas de CIs

standard cells ou mesmo a capacidade lógica de matrizes de transistores pré-difundidas (*gate arrays*).

No caso dos PLDs, independentemente do critério utilizado para o cálculo da sua capacidade lógica máxima, um determinado circuito digital é implementado apresentando uma certa taxa de utilização do componente. O produto capacidade x utilização define, a princípio, a complexidade do circuito implementado. Nas Tabelas 1 e 2 são apresentados dados sobre alguns componentes da Altera e da Xilinx [6] [7], com ênfase na

capacidade lógica máxima fornecida pelo respectivo fabricante.

Esta forma de definição de capacidade lógica não se apresenta tão precisa e adequada como no caso das matrizes *gate arrays*. Acredita-se que, devido à realização de manipulações lógicas pelos softwares de programação dos PLDs, é difícil saber, a priori, qual a complexidade resultante de um circuito após a sua compilação num componente PLD, em relação ao seu diagrama esquemático.

Tabela 1 - Características de PLDs da Altera Corp.

Família	Componentes	Macro células (= nº flip-flops)	Capacidade em portas equivalentes*
Clássica	EP610	16	0,6K
	EP910	24	0,9K
	EP1810	48	1,8K
Max 5000	EPM5016	16	0,6K
	EPM5032	32	1,2K
	EPM5064	64	2,5K
	EPM5128	128	5,0K
	EPM5130	128	5,0K
	EPM5192	192	7,5K
Max 7000	EPM7032	32	1,2K
	EPM7064	64	2,5K
	EPM7096	96	3,6K
	EPM7128	128	5,0K
	EPM7160	160	6,4K
	EPM7192	192	7,5K
	EPM7256	256	10,0K

(*) A quantidade de portas equivalentes utilizáveis é de 50%.

Tabela 2 - Características de PLDs da Xilinx, Inc.

Família	Componentes	CLBs	Células tri-state	Flip-flops	Capacidade (portas equiv.)
XC2000	XC2064	64	-	122	0,8K - 1,0K
	XC2018	100	-	174	1,2K - 1,5K
XC3000	XC3020	64	16	256	1,3K - 1,8K
	XC3030	100	20	360	2,0K - 2,7K
	XC3042	144	24	480	2,5K - 3,7K
	XC3064	224	32	688	4,0K - 5,5K
	XC3090	320	40	928	5,0K - 7,5K

Outras características que dificultam a obtenção da estimativa em portas equivalentes dos PLDs são:

a) **A utilização irregular das células lógicas (CLs):** os blocos lógicos configuráveis CLBs (Xilinx) ou macrocélulas (Altera), responsáveis pela implementação das funções lógicas existentes nas arquiteturas, podem ser utilizados em toda a lógica disponível, como também apenas implementar uma equação booleana muito simples, desperdiçando o restante da capacidade da CL, sendo também muito difícil manter uma regularidade no fator de utilização destes blocos.

b) **As portas lógicas básicas podem utilizar uma quantidade considerável da capacidade de uma CL:** uma porta OR exclusiva de 4 entradas, por exemplo, que corresponde a uma célula básica em bibliotecas *standard cells* e *gate arrays* (cerca de 2 a 3 portas equivalentes), utiliza toda uma macrocélula quando implementada nos componentes da Altera, devido a sua decomposição na forma de soma-de-produtos.

c) **O uso limitado de flip-flops e células tri-state devido à arquitetura fixa dos PLDs:** além de considerar a capacidade em portas equivalentes, deve-se observar também o número de flip-flops e células *tri-state* exi-

gidos pelo ASIC, pois os componentes PLDs apresentam um número limitado destas estruturas em sua arquitetura.

A par destas considerações, foram feitos estudos para tentar relacionar o critério de cálculo de complexidade de um ambiente *gate array* e dos PLDs. Inicialmente, utilizou-se circuitos funcionais desenvolvidos durante a avaliação funcional do GA2500 [11], e numa segunda fase, elaborou-se circuitos *benchmarks* específicos para este estudo comparativo.

4. ANÁLISE INICIAL: CIRCUITOS FUNCIONAIS

A estratégia inicial deste estudo foi aproveitar circuitos já projetados, de complexidade lógica bem conhecida, implementando-os em alguns PLDs (vide Tabela 3). Foram utilizados os três circuitos descritos em [11]:

- **MUL_4x4** - multiplicador de 4 x 4 bits em arquitetura *pipeline*: realiza a multiplicação binária em quatro estágios de processamento (quatro ciclos de relógio);

- **ULA_16** - unidade lógica e aritmética de 16 bits: realiza 6 operações básicas com dois vetores de 16 bits (soma, subtração, AND, OR, OR exclusivo e complemento);

Tabela 3 - Circuitos funcionais para análise inicial

Circuito	Complexidade (portas equiv.)	% lógica seqüencial	% lógica combinacional	Flip-flops	Células tri-state
QUIM	265	80,2	19,8	47	8
MUL_4x4	439	63,5	36,5	41	-
ULA_16	654	19,5	80,5	32	-
Ouvidor	1457	25,0	75,0	65	32

- **Ouvidor** - circuito "observador" de barramentos que auxilia na testabilidade de sistemas digitais através da implementação de equações polinomiais.

O quarto circuito funcional utilizado provém de um projeto realizado a nível de *standard cell* [12]: **QUIM** - parte

digital de uma interface para um sistema de aquisição de dados de sensores químicos.

Os resultados da compilação destes circuitos dentro dos componentes da Altera e da Xilinx são apresentados na Tabela 4.

Tabela 4 - Compilação dos circuitos funcionais nos PLDs.

Circuitos	Max5000	Max7000	Max2000*	Max3000*
QUIM	EPM5128 (43%) 1075 p.e.	EPM7096 (59%) 1062 p.e.	-	XC3020 (39 CLBs, 47 ff, 8 tri-state) 792 p.e.
MUL_4x4	EPM5064 (73%) 913 p.e.	EPM7096 (52%) 936 p.e.	XC2064 (43 CLBs, 41 ff) 538 p.e.	XC3020 (41 CLBs, 41 ff) 833 p.e.
ULA_16	EPM5192 (83%) 3113 p.e.	EPM7160 (92%) 2944 p.e.	XC2018 (80 CLBs, 32 ff) 960 p.e.	XC3030 (72 CLBs, 32 ff, 1440 p.e.
Ouvidor	EPM5130 (75%) 1875 p.e.	EPM7128 (76%) 1900 p.e.	-	XC3064 (133 CLBs, 65 ff, 32 tri-state) 2375 p.e.

Obs.: Abreviações - p. e. = portas equivalentes; ff = número de flip-flops utilizados.

(*) A complexidade em portas equivalentes nos PLDs da Xilinx foi avaliada através do número de CLBs utilizadas, tomando-se o melhor valor da faixa apresentada na tabela 2.

Observa-se, primeiramente, que para ambos os fabricantes, Altera e Xilinx, a **utilização da capacidade lógica máxima dos PLDs** empregados não corresponde a complexidade dos circuitos definida no ambiente *gate array* (Tabela 3), não se verificando nem mesmo uma proporção entre ambas.

A título de exemplo, para os circuitos **QUIM** e **MUL_4x4**, não foi possível utilizar o componente **EPM7064**, que possui capacidade lógica utilizável estimada de 1250 portas, embora este valor fosse numericamente suficiente. Também, os circuitos **ULA_16** e **Ouvidor** utilizaram mais do que 50% dos CLBs disponíveis nos componentes Xilinx **XC3030** e **XC3064**, respectivamente, o que corresponde a complexidades estimadas superiores às da tabela 3.

Notou-se que o **aproveitamento dos PLDs** ocorre de acordo com as características funcionais dos circuitos (síncrono, assíncrono, combinacional) e com o tipo de portas lógicas utilizadas nos diagramas esquemáticos (funções baseadas em lógica combinacional ou em tri-state). Como exemplo, o circuito **ULA_16**, que possui

uma complexidade bem menor do que o **Ouvidor** (Tabela 3), teve que utilizar PLDs maiores quando compilado pelo ambiente da Altera. No caso dos circuitos **MUL_4x4** e **QUIM**, de complexidade distinta, tiveram uma utilização bastante semelhante de CLBs pelo ambiente da Xilinx.

Deve-se chamar a atenção, também, quanto às **limitações do número de células tri-state e flip-flops** nestes componentes, particularmente para os componentes Xilinx: os circuitos **QUIM** e **Ouvidor** não puderam ser compilados na família **XC2000** por esta não possibilitar o uso de células lógicas *tri-state*; a mesma limitação fez com que o **Ouvidor** tivesse que ser implementado no componente **XC3064**, embora o **XC3042** apresentasse um número suficiente de CLBs para este circuito.

5. ANÁLISE COMPLEMENTAR: BENCHMARKS ESPECÍFICOS

O uso de apenas quatro circuitos funcionais não permite uma amostragem significativa para esta comparação de capacidade lógica entre uma matriz *gate array* de 2500 portas lógicas equivalentes e os PLDs da Altera

e da Xilinx. O fato de serem circuitos funcionais em média pequenos e que não cobrem certas aplicações, como máquinas de estados, pode levar a resultados tendenciosos.

A segunda fase deste trabalho consistiu na aplicação dos princípios de comparação implantados pelo PREP (*Programmable Electronics Performance Corporation*). O PREP é uma organização formada pelas empresas com interesse em componentes PLDs e ferramentas de CAD (*Computer Aided Design*) para estes ambientes. Tem por objetivo principal realizar comparações de capacidade e desempenho entre estes componentes e respectivos ambientes de projeto, a fim de esclarecer ao projetista dúvidas quanto à eficiência de arquiteturas e dos elementos de programação (fusíveis, EPROM e semelhantes) [9] [10].

Porém, a comparação realizada pelo PREP não questiona a medida de complexidade dos PLDs através da contagem de portas equivalentes, tal como apresentado na Tabela 4, apesar dos fabricantes fornecerem através desta contagem o tamanho dos seus componentes.

A filosofia de criação dos *benchmarks* pelo PREP é utilizar circuitos específicos, tanto combinacionais

quanto síncronos ou mistos, como contadores, máquinas de estado, acumuladores, unidades aritméticas ou mesmo *data paths*. Estes circuitos possuem uma complexidade baixa (entre 100 e 300 portas equivalentes) e são repetidos o maior número de vezes dentro do componente.

Assim, é possível conhecer a eficiência da utilização dos PLDs, segundo algumas topologias de circuito de interesse, que expressam características operacionais típicas às existentes em aplicações reais.

A estratégia utilizada nesta etapa de estudos foi desenvolver 8 *benchmarks*, segundo as especificações do PREP, e implementá-los numa matriz *gate array*, pelo princípio de repetição, para então definir os circuitos a serem compilados nos PLDs, conforme Tabela 5. É importante enfatizar, novamente, que não está sendo avaliada a capacidade de roteamento de cada ambiente, sendo este um estudo a parte.

Os resultados das implementações dos circuitos bench1 a bench8 (circuitos resultantes da repetição dos *benchmarks*) podem ser vistos na Tabela 6.

Tabela 5 - Benchmarks específicos para comparação de capacidade lógica

Benchmarks	Complexidade* (portas equiv.)	Nº de repetições	Circuito resultante (complexidade)
data path	160,0 (56% / 44%)	15	bench 1 (2400 p. e.)
timer	283,2 (47% / 53%)	8	bench2 (2266 p.e.)
máquina de estado (8 bits)	94,0 (18% / 82%)	26	bench3 (2444 p.e.)
máquina de estado (16 bits)	205,2 (11% / 89%)	12	bench4 (2462 p.e.)
circuito aritmético	246,4 (18% / 82%)	10	bench5 (2464 p.e.)
acumulador de 16 bits	89,6 (100% / 0%)	27	bench6 (2419 p.e.)
contador up	237,2 (38% / 62%)	10	bench7 (2372 p.e.)
contador down	225,2 (40% / 60%)	11	bench8 (2477 p.e.)

(*) Os valores percentuais correspondem às porcentagens de lógica seqüencial e combinacional do circuito, respectivamente.

Tabela 6 - Resultados dos *benchmarks* nos PLDs.

Circuitos	Max5000*	Max7000*	XC3000*
bench1	EPM5128 (72%) EPM5192 (76%) 4650 p.e.	EPM7256 (93%) 4650 p.e.	XC3090 (254 CLBs, 240 ff) 3969 p.e.
bench2	-	EPM7128 (81%) EPM7192 (89%) 5363 p.e.	XC3090 (235 CLB's, 192 ff) 3672 p.e.
bench3	-	-	XC? (508 CLBs, 78 ff) 7938 p.e.
bench4	-	-	-
bench5	-	-	XC? (341 CLBs, 80 ff) 5328 p.e.
bench6	-	-	XC3064 (219 CLBs, 432 ff) 3911 p.e.
bench7	EPM5192 (83%) 3113 p.e.	EPM7192 (85%) 3188 p.e.	XC3090 (300 CLBs, 160 ff) 4688 p.e.
bench8	EPM5192 (91%) 3413 p.e.	EPM7256 (71%) 3550 p.e.	XC? (329 CLBs, 176 ff) 5141 p.e.

(*) Em alguns circuitos foi realizado o particionamento automático do ambiente Altera.

(**) O critério para contagem do número de portas equivalentes é o mesmo utilizado na tabela 3.

A análise destes resultados é surpreendente. Poder-se-ia dizer que todos os circuitos *benchs* apresentam uma complexidade aproximada de 2500 portas equivalentes. Nota-se que apenas os circuitos *bench1*, *bench2*, *bench6* e *bench7* puderam ser implementados nos componentes da família XC3000 (XC3090).

Os circuitos *bench1*, *bench2* e *bench6* para Max5000 e *bench2* e *bench6* para Max7000 não couberam em nenhum elemento das respectivas famílias. E os circuitos *bench1*, *bench7* e *bench8* couberam apenas nos dois maiores elementos da família Max7000 (EPM7000).

Particularmente, os circuitos *bench3*, *bench4* e *bench5* tiveram dificuldades de serem compilados pelo ambiente da Altera, devido a densidade de lógica combinacional; o sintetizador lógico não conseguiu realizar seu trabalho.

Para estes *benchs*, realizou-se a compilação do bloco básico dos *benchmarks* e fez-se uma interpolação

da ocupação final em relação ao número de repetições, cujo resultado revelou a impossibilidade de implementação. Problema semelhante foi encontrado na compilação do *bench4* dentro do ambiente da Xilinx (para a família XC3000), quando se utiliza o software para plataforma PC.

6. CONCLUSÕES

Foi proposta uma metodologia de análise da capacidade lógica de PLDs, em relação a contagem de portas lógicas equivalentes utilizadas em *gate arrays*. Alguns projetos de circuitos digitais, já realizados segundo a metodologia *gate array*, para uma matriz de 2500 portas equivalentes, foram implementados em EPLDs da Altera e FPGAs da Xilinx, tendo sido comparada a utilização dos componentes PLDs e a complexidade aparentes destes circuitos.

Os resultados preliminares indicam uma surpreendente discrepância entre a contagem de portas lógicas nas metodologias de projeto *gate array* e PLD, inclusive, não havendo uma proporção entre estes valores.

Circuitos digitais de cerca de 2500 portas em *gate array* podem atingir no ambiente Altera contagens estimadas na faixa de 3113 a 5363 portas equivalentes, correspondendo a um aumento de 25% a 115%. No ambiente Xilinx, obteve-se contagens estimadas na faixa de 3672 a 7938, ou seja, 47% a 217%.

Como trabalhos futuros, pretende-se estender esta análise para comparações de desempenho elétrico, estudos sobre a influência da intervenção do projetista

sobre a alocação de CLs e roteamento de interconexões, durante a fase de compilação do circuito.

Outro aspecto importante diz respeito ao caminho inverso de implementação, partindo-se de circuitos em PLDs para sua integração em *gate arrays*. Resultados preliminares, obtidos da análise de circuitos desenvolvidos pelo CPqD/TELEBRÁS para o sistema Trópico, que foram descritos em VHDL e prototipados em FPGAs da Xilinx, são reportados na Tabela 7 abaixo [13] [14].

Claramente, observa-se que a complexidade de tais circuitos, quando implementados num *gate array* de 2500 portas lógicas equivalentes, sofre uma redução de cerca de 60%. Tais resultados serão mais profundamente analisados em trabalhos futuros.

Tabela 7 - Comparação de circuitos inicialmente implementados em PLDs.

Circuito	Complexidade estimada XC3042 (portas equivalentes)	Complexidade de um gate array de 2500 (portas equivalentes)	% de redução
TB45	3267	1298	60,3
TB46	2780	1190	57,2

Portanto, conclui-se que, do ponto de vista da engenharia de sistemas, implementar ASICs por meio de *gate arrays* ou PLDs resulta em avaliações de complexidade totalmente distintas e não coerentes entre si. Um sistema cuja complexidade seja avaliada por sua implementação em ambiente FPGA, por exemplo, pode ser mais facilmente realizável em *gate array*, ou mesmo *standard cell*, com benefícios óbvios de custo e desempenho.

Por fim, dado o contexto do Projeto Conjunto de Hardware e Software, tem-se uma avaliação preliminar das ferramentas de implementação de ASICs para prototipagem rápida, as quais serão consideradas como alternativas para a definição de um sistema de projeto HW/SW, ora em fase de investigação.

AGRADECIMENTOS

O autor agradece às empresas *Hicad Sistemas Ltda e União Digital Com. Rep. Ltda*, em especial a *Vanderlei Perez Sanches e José Carlos Stagni*, pelo empréstimo dos softwares de PLDs, e à colaboração dos engenheiros do CPqD/TELEBRÁS *Luiz M. A. Madureira*,

Carlos G. Kruger e Alexandre Soares Pires, que tornaram possível a realização deste trabalho. Também ao engenheiro *Renato Peres Ribas*, que conduziu substancial parte dos trabalhos a título de tese de Mestrado, sob suporte do Programa RHA/E. Agradecemos à Fundação Centro Tecnológica para Informática, FCTI, onde as trabalhos preliminares foram conduzidos, e pelo suporte de laboratórios e infraestrutura, sem os quais não seriam atingidos os objetivos relatados.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Hardware/Software codesin, D & T Roundtable, IEEE Design & Test of Computers, pp. 83-91, Março, 1993.
- [2] D. E. THOMAS, J. K. ADAMS, H. SCHMIT, A model and methodology for hardware-software codesign, IEEE Design & Test of Computers, pp. 6-15, Setembro, 1993.
- [3] R. K. GUPTA, Program implementation schemes for hardware-software systems, IEEE Computer, pp. 48-55, Janeiro, 1994.

- [4] REIS, A. I., GÜNTZEL, J. L. & RIBAS, R. P. Algumas formas de implementação de ASICs. Anais do VII Simpósio Brasileiro de Concepção de Circuitos Integrados, pp. 15-34, Rio de Janeiro - RJ, Setembro, 1992.
- [5] Functional flexibility keeps ASICs in demand. *Journal of Electronic Engineering - JEE*. January, 1993. Vol. 30, Nº 313, Dempa Publications. pp. 66-69.
- [6] Data Book da Altera Corp., 1993. San Jose, Califórnia.
- [7] The Programmable Logic Data Book, 1993. Xilinx, Inc. San Jose, Califórnia.
- [8] SIMÕES, S. A.; CHAVEZ, F.; RIBAS, R. P.; FINCO, S.; CUIN, M. & BEHRENS, F. H. Matriz Gate Array CMOS avançada, configurável por um único nível de metal. Anais do VII Congresso da SBMicro, pp. 281-291, São Paulo - SP, Julho, 1992.
- [9] A Systems Perspective on Inter-pretng the PREP Benchmarks. Documento de divulgação da Xilinx, Inc.
- [10] PREP Benchmarks For Pro-grammable Logic Devices (White Paper). Documento de divulgação da Altera Corp.
- [11] Estruturas e Circuitos de Teste do Projeto GA2500. Documento Interno 011/93 LPCI-IM-CTI. 1993.
- [12] CHAVEZ, F.; FINCO, S.; ROHWEDDER, J. J. R. & PASQUINI, C. Integrated multichannel voltammetric detection system: design description. Anais do VIII Congresso da SBMicro, pp. XII. 34-36, Campinas - SP, Setembro, 1993.
- [13] Documentação de Projeto do TB45. Documento interno do CPqD / TELEBRÁS.
- [14] Documentação de Projeto do TB46. Documento interno do CPqD / TELEBRÁS.