

# Síntese Direta de Circuitos Seqüenciais Assíncronos VLSI Robustos de Baixo Consumo

Duarte L. Oliveira

Divisão de Engenharia Eletrônica do Instituto Tecnológico de Aeronáutica – IEAA – ITA  
Praça Marechal Eduardo Gomes, 50 - CEP 12228-900 - São José dos Campos - São Paulo - Brasil.

**Resumo** — Muitas aplicações no ambiente aeroespacial requerem dos circuitos digitais VLSI um desempenho crítico. Este desempenho está relacionado com alta capacidade de integração, alta velocidade, baixo consumo de energia, robustos aos efeitos radiativos, baixa interferência eletromagnética. Circuitos seqüenciais VLSI síncronos são bastantes populares para os projetistas de VLSI, mas possuem problemas para obter este desempenho. Neste artigo propomos um método prático para projetar circuitos seqüenciais VLSI no paradigma assíncrono. Este paradigma tem um maior potencial de satisfazer o desempenho das aplicações aeroespaciais quando comparado com os circuitos seqüenciais VLSI síncrono.

**Palavras chaves** — Lógica assíncrona, risco, corrida crítica, máquinas de estado finito, quase independente de atraso.

## I. INTRODUÇÃO

Circuitos digitais VLSI quando aplicados no ambiente aeroespacial requerem alta capacidade de integração, alta velocidade, baixo consumo de energia, robustos aos efeitos radiativos, baixa interferência eletromagnética e serem testados exaustivamente [1]. Um caminho que satisfaz a alta capacidade de integração e alta velocidade é a tecnologia MOS *Deep-Sub-Micron* (por exemplo: 70nm, 500M transistores por chip e  $f=2,5\text{GHZ}$ ) [2]. Essa tecnologia necessita operar com baixo ruído, baixo consumo de potência e a diferença entre o atraso máximo e mínimo nas linhas e portas é maior quando comparado com outras tecnologias MOS. Os circuitos seqüenciais MOS-DSM síncronos usam um sinal de relógio global para efetuar as suas operações e são bastantes populares devido à simplicidade de projeto. Também há uma oferta abundante de ferramentas CAD comerciais. Um sério problema na tecnologia MOS-DSM é conviver com o sinal de relógio global, porque ele é um grande causador de ruído e de alta emissão eletromagnética, consome uma parte significativa da potência e definir a distribuição do sinal de relógio é uma tarefa de complexidade crescente (por exemplo: *skew clock*). A análise de temporização de circuitos digitais MOS-DSM síncronos de alta integração é extremamente difícil [3].

Circuitos seqüenciais assíncronos operam por eventos não necessitam de um sinal de relógio, portanto eles têm: baixo ruído, baixo consumo de potência, baixa interferência eletromagnética e não possuem problemas de *skew clock* [4]. Existem diferentes classes de circuitos assíncronos [5]. A classe é definida pelo modo de operação (interação com o ambiente) e o modelo de atraso em que o circuito opera corretamente [5].

Duarte Lopes de Oliveira, duarte@ita.br, Tel +55-12-39746813, Ramal 6813, Fax +55-12-39746930.

Uma classe de circuitos seqüenciais assíncronos denominada quase independente de atraso (*Quasi-Delay-Insensitive* – QDI) opera no modo entrada/saída e satisfaz o modelo de portas e linhas com atrasos indefinidos, mas finitos. O modelo assume a restrição *isochronik fork* onde as linhas com *fan-out*  $>1$  (desvio) possuem o mesmo atraso [5,6]. Na prática a classe QDI é equivalente à classe independente da velocidade (*Speed-Independent*) [7,8]. A Classe QDI possui outras características que são importantes no ambiente aeroespacial quando comparado com os circuitos síncronos [5,9,10]: **a)** potencial de um melhor tempo de latência (velocidade dos sinais de saída); **b)** robusto a radiação do tipo *Single-Event Upset* (SEU) e faltas do tipo atraso e do tipo *Stuck-at* (classes de faltas facilmente testáveis); **c)** alta modularidade permitindo maior reusabilidade; **d)** robusto a variações de temperatura e de tensão de alimentação; **e)** melhor desempenho no projeto de sistemas seguros (por exemplo, criptografia).

O problema dos circuitos seqüenciais assíncronos está na dificuldade do projeto, porque eles devem ser livres de risco (*hazards*) e corrida crítica [11]. Outros problemas são a falta da cultura assíncrona nos projetistas de VLSI e principalmente a falta de ferramentas CAD comerciais [12]. Nos últimos anos diversos produtos comerciais de alto desempenho foram implementados com circuitos assíncronos, entre as quais, podemos citar [13]: microcontrolador 8051 e *pager* da Philips; processador multimídia da Sharp; cartão inteligente de contato da France Telecom e o microcontrolador acadêmico SPEAR que é voltado para aplicações embarcadas em ambientes de tempo real.

Neste artigo propomos uma técnica de síntese voltada para circuitos seqüenciais assíncronos QDI. O nosso método sintetiza os nossos circuitos diretamente da especificação popular denominada modo rajada (E\_MR – *Burst-Mode Specification*) e implementa na arquitetura alvo *gC standard* (*full-custom* pseudo-estático) [14,15]. A E\_MR é um grafo de transição de estados que é muito familiar aos projetistas do mundo síncrono. Os nossos circuitos pertencem à classe popular de máquinas de estado finito assíncrona denominada modo rajada (MEF\_MR). Os estados são codificados no código *pseudo-Gray*, permitindo redução de energia. As grandes vantagens do nosso método é que não há necessidade do conhecimento de toda a teoria do projeto livre de risco para os circuitos seqüenciais assíncronos QDI e para muitas aplicações o projeto pode ser realizado manualmente, sem a necessidade de ferramentas. O nosso método propicia na implementação *gC standard* pequenas pilhas de transistores PMOS e NMOS [15]. Os nossos resultados em área (número de transistores) são bastante satisfatórios.

Este artigo está estruturado na seguinte forma: Na seção II um resumo da especificação modo rajada e os diferentes rótulos para cada transição de estado da E\_MR; na seção III apresentamos o nosso método; na seção IV ilustramos o nosso método com um exemplo da literatura; na seção V discutimos as vantagens e desvantagens do nosso método e apresentamos alguns resultados; na seção VI finalmente apresentamos as nossas conclusões.

## II. ESPECIFICAÇÃO MODO RAJADA

Inicialmente, a E\_MR foi introduzida em um projeto da HP [16], e posteriormente formalizada em [14]. Ela é descrita por um grafo, contém um número finito de estados (vértices) e de transições de estado (arcos), sendo que um estado é inicial. Cada transição de estado é rotulada por um conjunto não vazio de sinais de entrada (entrada rajada) e um conjunto de sinais de saída (saída rajada) que pode ser vazio. Para um estado, quando todas as entradas de alguma entrada rajada, mudarem seus valores, o circuito gera a saída rajada correspondente, e assume um novo estado. Somente as mudanças de entrada especificadas podem ocorrer. As transições dos sinais de entrada (entrada rajada) são permitidas mudar monotonicamente em qualquer ordem, e em qualquer tempo. Um sinal é rotulado na transição de estado como s+ ou s-, que significam respectivamente as transições 0→1 ou 1→0 do sinal s. Um exemplo de E\_MR está mostrado na Fig. 1. Esta especificação descreve um simples controlador, tendo 4 entradas (a,b,c,d) e 3 saídas (x,y,z).

Na E\_MR todos os sinais de entrada e de saída devem obedecer à *seqüência de polaridade*, por exemplo: ..a+, a-, a+.... ou ....a-, a+, a-.... seqüências válidas. Seqüências não válidas ...a+,a+,a-... ou ...a-, a-,a+....

Há duas restrições adicionais na descrição da E\_MR [14]: **a)** nenhuma entrada rajada nas transições de estado que emergem de um mesmo estado pode ser um subconjunto de uma outra entrada rajada deste estado, isto é todo estado no E\_MR deve ter a propriedade do *subconjunto máximo* (ver Fig. 2); **b)** transições de estado que incidem para um mesmo estado devem gerar o mesmo conjunto de valores de entrada, isto é, todo estado no E\_MR deve ter a propriedade do *ponto de entrada única* (ver Fig. 3). A E\_MR voltada para QDI sofre uma restrição, a saída rajada não pode ser vazia.

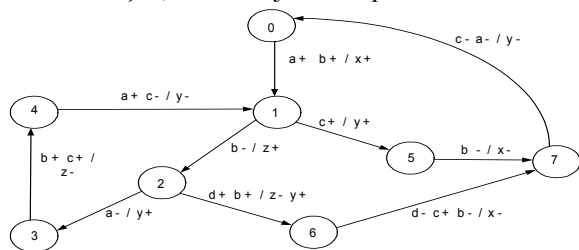


Fig. 1. Especificação modo rajada.

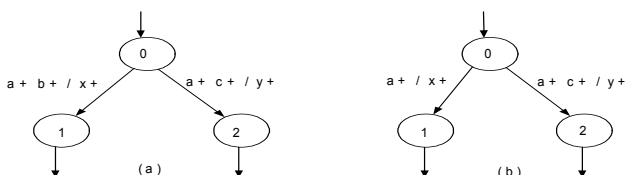


Fig. 2. Propriedade subconjunto máximo: a) válida; b) inválida.

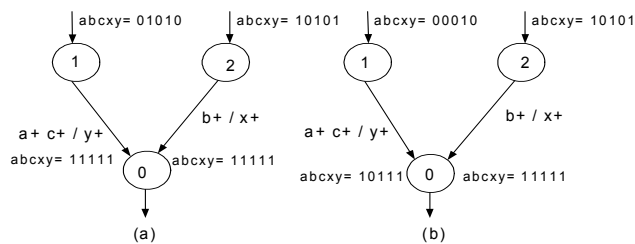


Fig. 3. Propriedade ponto de entrada única: a) válida; b) inválida.

### A. Grafo de transição estado-sinal

O nosso método trabalha diretamente na E\_MR, mas usa diferentes rótulos nas transições de estado da E\_MR. Para fins formais os diferentes rótulos nas transições de estado levam a uma especificação similar ao E\_MR que é denominada neste artigo de grafo de transição de estado-sinal (GTES). Para cada sinal não entrada (sinais de saída e de estado) do E\_MR gera-se os grafos GTES<sub>SET</sub> e GTES<sub>RESET</sub>. A geração dos rótulos no GTES segue duas regras:

**Regra 1:** sinal não entrada  $X \in E\_MR \rightarrow GTES_{X-SET}$

1. Transição de estado da E\_MR, rotulado X (0→1), no GTES<sub>X-SET</sub> o rótulo é um produto (entrada rajada).
2. Transição de estado da E\_MR, rotulado X (1→1), no GTES<sub>X-SET</sub> o rótulo é *don't-care* {\*}
3. Transição de estado da E\_MR, rotulado ou X(0→0) ou X(1→0), no GTES<sub>X-SET</sub> o rótulo é zero.

**Regra 2:** sinal não entrada  $X \in E\_MR \rightarrow GTES_{X-RESET}$

1. Transição de estado da E\_MR, rotulado X (1→0), no GTES<sub>X-RESET</sub> o rótulo é um produto (entrada rajada).
2. Transição de estado da E\_MR, rotulado X (0→0), no GTES<sub>X-RESET</sub> o rótulo é *don't-care* {\*}
3. Transição de estado da E\_MR, rotulado ou X(1→1) ou X(0→1), no GTES<sub>X-RESET</sub> o rótulo é zero.

Para ilustrar a aplicação das regras 1 e 2, as Fig. 4 e 5 mostram respectivamente o GTES<sub>Y-SET</sub> e GTES<sub>Y-RESET</sub> do sinal não entrada Y do E\_MR da Fig. 1.

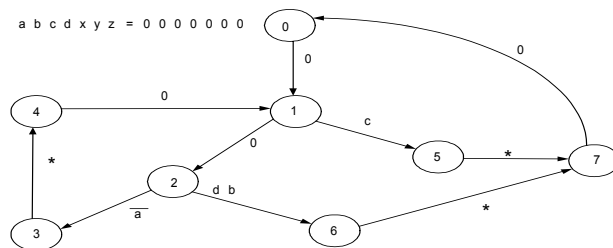


Fig.4. Grafo de transição estado-sinal: sinal Y<sub>SET</sub>.

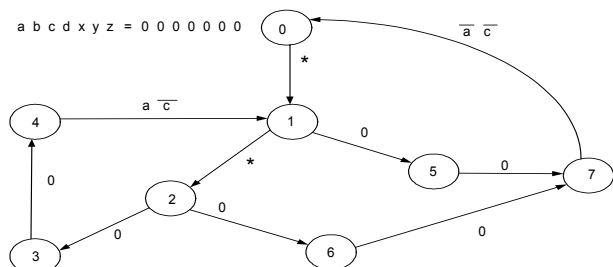


Fig. 5. Grafo de transição estado-sinal: sinal Y<sub>RESET</sub>.

### III. MÉTODO DE SÍNTESE: PROCEDIMENTO

O nosso método é composto por 4 passos, parte da  $E\_MR$  e gera para cada sinal não entrada as equações de dois níveis  $E\_SET$  e  $E\_RESET$  da arquitetura  $gC$  standard (Elemento C generalizado – ver Fig. 6):

1. Codificar o grafo de transição de estados ( $E\_MR$ ) usando as regras propostas na seção A.
2. Para cada sinal não entrada gerar o grafo de transição estado-sinal (GTES).
3. Para o GTES encontrar todas as regiões ativas usando o procedimento proposto na seção B.
4. Para o GTES obter as equações  $E\_SET$  e  $E\_RESET$  de dois níveis usando os dois lemmas propostos na seção C. Estas equações são transformadas em pilhas de transistores NMOS e PMOS.

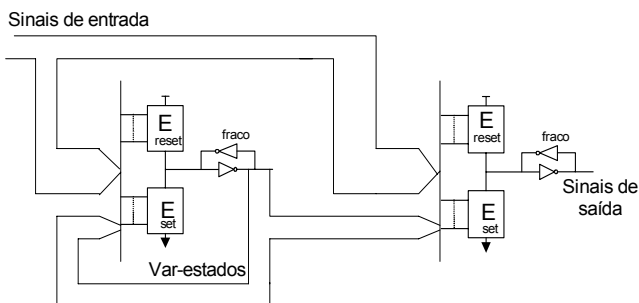


Fig. 6. Arquitetura alvo:  $gC$  standard.

#### A. Codificação pseudo-Gray

Diversos algoritmos de codificação livre de corrida crítica podem ser usados na codificação da  $E\_MR$  voltado para a classe de MEF-MR QDI [17,18]. Neste artigo propomos um algoritmo heurístico que pela sua simplicidade pode ser aplicado manualmente. O nosso algoritmo tem como a principal preocupação a redução da atividade de chaveamento dos sinais de estado.

A Partir do estado inicial da  $E\_MR$  percorrer por profundidade a  $E\_MR$  e obter todos os caminhos. Um caminho é uma seqüência de transições de estado que inicia e termina em um mesmo estado da  $E\_MR$ . Analisar se há caminhos onde todos os estados estão contidos em algum outro caminho. Caso afirmativo, eliminar tal caminho obtendo assim o conjunto de caminhos mínimos (CCM). Todos os caminhos pertencentes ao CCM devem ter um número par de estados. Caso algum caminho não tenha, inserir um estado “ponte”. Os estados pertencentes a cada caminho são codificados na seqüência de Gray.

O número de sinais de estado ( $N_{S\_EST}$ ) necessário para a codificação *pseudo-Gray* livre de corrida crítica da  $E\_MR$  é  $N_{S\_EST} = N_C + N_{S\_EST-C}$  e  $N_{S\_EST-C} = \lceil \log_2 N_{EC} \rceil$ , onde  $N_C$  é o número mínimo de caminhos,  $N_{EC}$  é o maior número de estados em um caminho e  $N_{S\_EST-C}$  é o número de sinais de estado do maior caminho. Finalmente, verificar se há sinais de estado não usados; caso afirmativo eliminá-los.

Como ilustração, a  $E\_MR$  da Fig. 1 tem os seguintes números:  $N_C=3$ ,  $N_{EC}=5$ ,  $N_{S\_EST-C}=3$  e  $N_{S\_EST}=6$ . Este exemplo usa 4 sinais de estado para a codificação (redução  $\rightarrow$  dois sinais de estados não usados). Fig. 7 mostra a  $E\_MR$  codificado livre de corrida crítica, onde as transições de estado estão rotuladas com o operador seqüência  $\{>\}$  que define o ciclo de máquina (entrada rajada  $\rightarrow$  variável de

estado  $\rightarrow$  saída rajada). Na transição de estado 6 $\rightarrow$ 7 da Fig. 7 há a inclusão do estado ponte (1010).

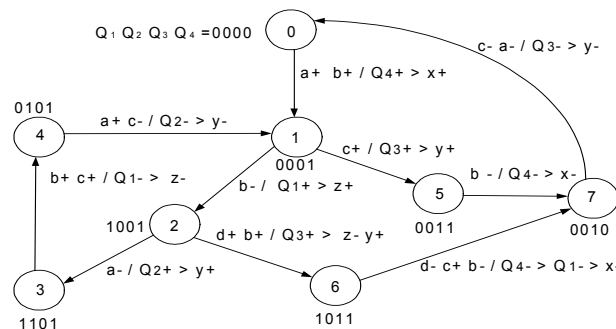


Fig. 7.  $E\_MR$  codificado livre de corrida.

#### B. Região ativa no GTSC

No  $GTES_{SET}$  ou  $GTES_{RESET}$  são detectadas regiões ativas do sinal não entrada. Uma região ativa  $RA_j$  (sinal não entrada) no GTES é formada pela seqüência de transições de estado rotuladas como produto e *don't-care*. Transições de estado que incidem no mesmo estado e são rotuladas com produto formam uma única região ativa. As Fig. 8 e 9 mostram respectivamente as regiões ativas do sinal Y da  $E\_MR$  da Fig. 1.

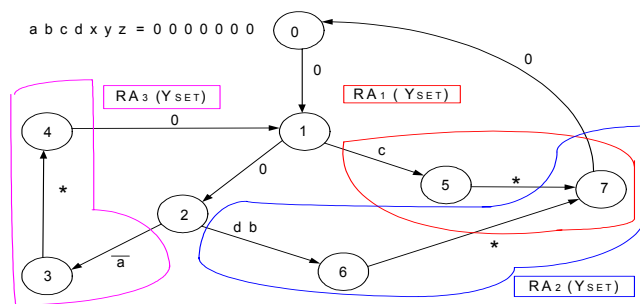


Fig. 8.  $GTES_{Y-SET}$ : 3 regiões ativas.

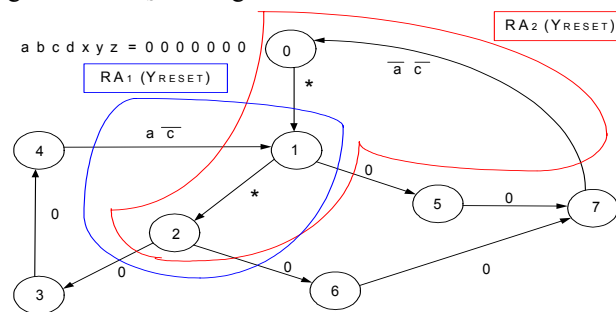


Fig. 9.  $GTES_{Y-RESET}$ : 2 regiões ativas.

#### C. Minimização lógica livre de risco

Os implicantes (produtos) aceitos das equações de dois níveis  $E\_SET$  e  $E\_RESET$  da arquitetura  $gC$  standard são obtidos através da aplicação dos lemmas 3.1 e 3.2 abaixo. Todo produto respectivamente pertencente à  $E\_SET$  ou  $E\_RESET$  deve satisfazer o lemma 3.1 ou o lemma 3.2. Diferentes procedimentos de minimização lógica podem ser adaptados para obter a cobertura mínima, por exemplo, algoritmo Quine-Mccluskey [19] e outros [20,21]. Para obter a cobertura mínima usamos uma variante do mapa de Karnaugh, conhecida como mapa K-reduzido.

**Lemma 3.1** (sem prova) Seja o sinal não entrada  $X \in E\_MR$ ,  $GTES_{X-SET}$ ,  $CRA_{X-SET}$  o conjunto de regiões ativas do  $GTES_{X-SET}$ , onde  $CRA_{X-SET} = \{RA_{1-X-SET}, RA_{2-X-SET}, \dots, RA_{K-X-SET}\}$  e  $CI_{X-SET}$  o conjunto de implicantes que cobre  $CRA_{X-SET}$ , onde  $CI_{X-SET} = \{IP_1, IP_2, \dots, IP_W\}$ . A cobertura  $CI_{X-SET}$  é livre de risco no modelo QDI se e somente se cada  $RA_{J-X-SET} \in CRA_{X-SET}$  é totalmente coberto por um único implicante  $IP_1 \in CI_{X-SET}$  e se houver algum outro implicante  $IP_F \in CI_{X-SET}$  que intercepta  $RA_{J-X-SET}$  então  $IP_F$  deve cobrir totalmente  $RA_{J-X-SET}$ .

**Lemma 3.2** (sem prova) Seja o sinal não entrada  $X \in E\_MR$ ,  $GTES_{X-RESET}$ ,  $CRA_{X-RESET}$  o conjunto de regiões ativas do  $GTES_{X-RESET}$ , onde  $CRA_{X-RESET} = \{RA_{1-X-RESET}, RA_{2-X-RESET}, \dots, RA_{W-X-RESET}\}$  e  $CI_{X-RESET}$  o conjunto de implicantes que cobre  $CRA_{X-RESET}$ , onde  $CI_{X-RESET} = \{IP_1, IP_2, \dots, IP_K\}$ . A cobertura  $CI_{X-RESET}$  é livre de risco no modelo QDI se e somente se cada  $RA_{J-X-RESET} \in CRA_{X-RESET}$  é totalmente coberto por um único implicante  $IP_1 \in CI_{X-RESET}$  e se houver algum outro implicante  $IP_F \in CI_{X-RESET}$  que intercepta  $RA_{J-X-RESET}$  então  $IP_F$  deve cobrir totalmente  $RA_{J-X-RESET}$ .

No mapa K-reduzido as variáveis de estado são declaradas externamente e os sinais de entrada são declarados internamente. Para ilustrar, aplicamos a nossa técnica de minimização lógica usando o mapa K-reduzido na Fig. 8. A Fig. 10 mostra o mapa K-reduzido do sinal Y para  $Y_{SET}$  com as três regiões ativas e os três implicantes que satisfazem o lemma 3.1. A Equação de dois níveis livre de risco é:

$$Y_{SET} = c Q1' Q3 Q4 + b d Q1 Q3 + a' Q2$$

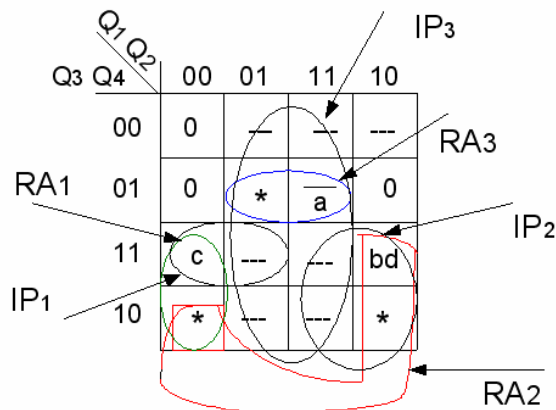


Fig. 10. Mapa K-reduzido para  $Y_{SET}$ .

IV. EXEMPLO

Nesta seção vamos ilustrar o nosso método com um exemplo da literatura. Fig. 11 mostra a  $E\_MR$  ALU1 [22]. Fig. 12 mostra o  $E\_MR$  codificado gerado pelo passo 1. A codificação está voltada para a classe QDI. O passo 2 gera o GTES. O passo 3 gera as GTES com as respectivas regiões ativas. Fig. 13 mostra o GTES $_{Q4-SET}$  com as regiões ativas. Fig. 13 mostra um tipo de interseção entre as regiões  $RA_1$  e  $RA_2$  envolvendo o produto  $EvDone' M1A$  na transição de estado  $6 \rightarrow 3$  que viola o modelo QDI. A solução é inserir o sinal  $Q4$  no produto  $EvDone' M1A Q4'$ . Finalmente o passo 4 obtém as equações  $E_{SET}$  e  $E_{RESET}$  para os sinais não entrada. Fig. 15 mostra o mapa K-reduzido para  $Prech_{SET}$ . As Fig. 16 e 17 mostram parcialmente o circuito resultante.

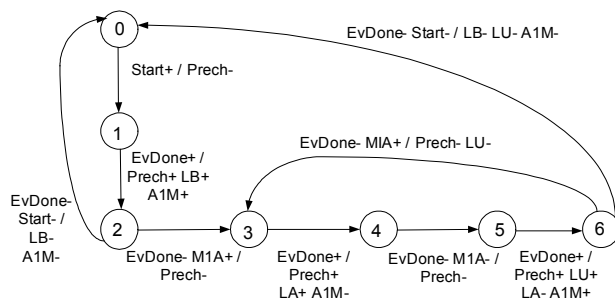


Fig. 11.  $E\_MR$  ALU1.

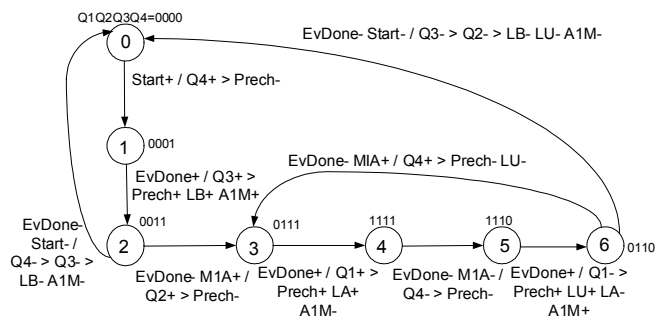


Fig. 12.  $E\_MR$  ALU1 codificado.

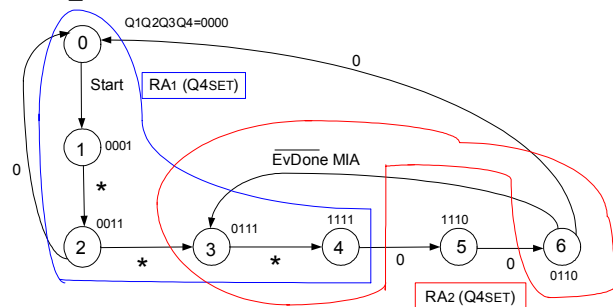


Fig. 13.  $GTES_{Q4-SET}$  ALU1: regiões ativas do sinal  $Q4_{SET}$ .

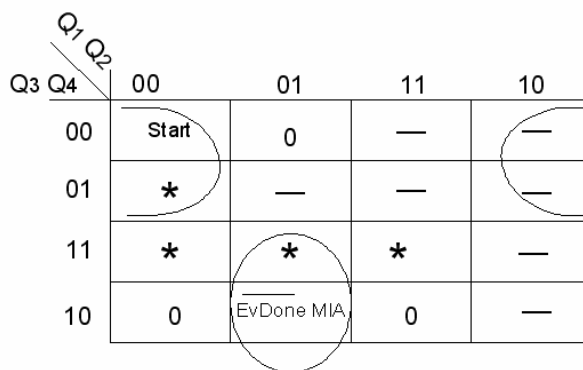


Fig. 14. Mapa K-reduzido:  $Q4_{SET}$ .

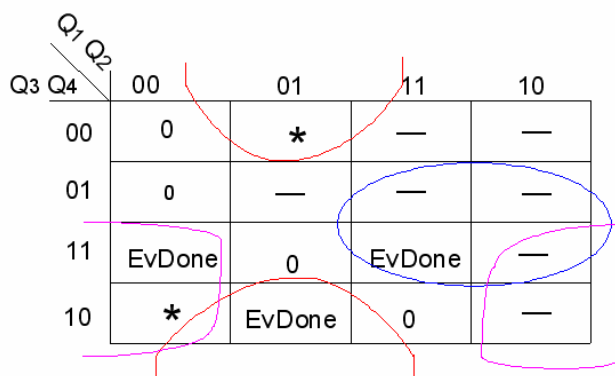


Fig. 15. Mapa K-reduzido:  $Prech_{SET}$ .

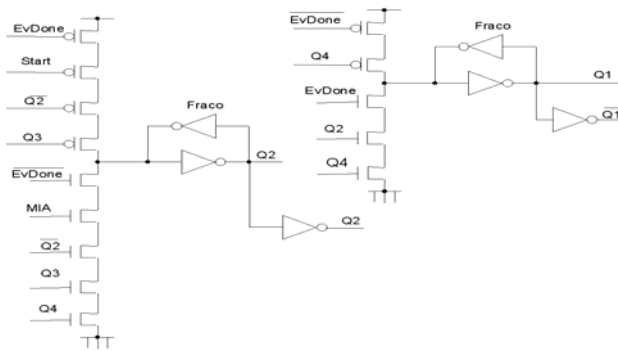


Fig. 16. Circuito transistorizado parcial ALU1: Q1 e Q2

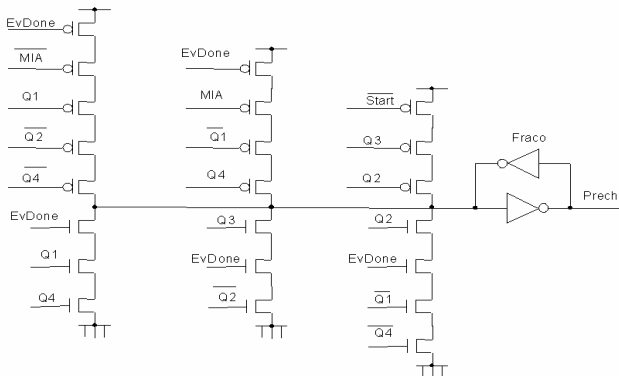


Fig. 17. Circuito transistorizado parcial ALU1: Prech

## V. DISCUSSÃO E RESULTADOS

As MEF\_MR QDI são potencialmente superiores comparando com as MEF síncronas. Diversas propriedades que as MEF\_MR QDI possuem mostram o seu grande potencial, principalmente quando aplicado no ambiente aeroespacial. Diversos métodos foram propostos para síntese das MEF\_MR, mas elas operam no modo fundamental generalizado, portanto são menos robustas [15,23,24]. Recentemente, síntese de MEF\_MR QDI foram propostas, mas não sintetizam MEF com muitos estados [25]. O nosso método permite sintetizar tais máquinas, porque toda a síntese é realizada na E\_MR e mapa K-reduzido. O nosso método não realiza minimização de estados e o passo de assinalamento de estados é realizado manualmente através de uma heurística (extrair os caminhos na E\_MR) e o uso da sequência de Gray. Mapas K-reduzido de 8 variáveis são possíveis de manipular manualmente, isto permite MEF de até 256 estados com um número qualquer de sinais de entrada.

Os exemplos (Fig. 1 e 11) sintetizados pelo nosso método respectivamente usaram 123 e 145 transistores. Comparando com a síntese automática do método [25] teve um aumento médio de 38% de transistores. Este aumento deve-se ao fato que os estados são representados somente por variáveis de estado e não por saídas e variáveis de estado.

## VI. CONCLUSÕES

Neste artigo trabalhamos com uma classe popular de circuitos assíncronos que são as máquinas de estado finito modo rajada. Estas máquinas operando na classe QDI podem ser muito úteis nas aplicações aeroespaciais. Elas têm um potencial para satisfazer um conjunto de requisitos críticos que as aplicações aeroespaciais possuem. Neste artigo

propomos um método para síntese desta classe de circuito que são implementados na forma *full-custom*, que proporciona alto desempenho. Como futuro trabalho, o desenvolvimento da ferramenta para a síntese automática das MEF\_MR QDI.

## REFERÊNCIAS

- [1] J. J. Wang, R. B. Katz, et al. "SRAM Based Re-programmable FPGA for Space Application," IEEE Trans, Nuclear Science, vol. 46, Nro 6, December, 1999, pp. 1728-1735.
- [2] D. Goldhaber-Gordon, et al., "Overview of Nanoelectronic Devices," Proc. of the IEEE, vol. 85, No. 4, April 1997, pp.521-540.
- [3] C. Constantinescu, "Trends and Challenges in VLSI Circuits Reliability," IEEE Micro, 23 (4), 2003.
- [4] S. B. Furber, "Breaking Step the Return of Asynchronous Logic," IEE Review, July 1993, pp.159-162.
- [5] Sparso, J. and Furbers, S., "Principles of Asynchronous Circuit Design — a System Perspective." Kluwer Academic Publishers, 2001.
- [6] Mayers, C. J. "Asynchronous Circuit Design," Wiley & Sons, Inc., 2001.
- [7] S. Hauck, "Asynchronous Design Methodologies: An Overview", Proc. of the IEEE, January 1995, Vol. 83:1 pp.69-93.
- [8] T. -A. Chu, "Synthesis of Self-Timed VLSI Circuits from Graph-Theoretic Specifications," PhD thesis, June, 1987, Dept. of EECS, MIT.
- [9] W. Jiang, W. and A. J. Martin "SEU-Tolerant QDI Circuits". Proc. of the 11th IEEE Int. Symposium on Asynchronous Circuits and Systems, pp.156-165, 2005.
- [10] J. DI, P. K. Lala and D. Vasudevan, "On the effect of Stuck-at Faults on Delay-insensitive Nanoscale Circuits," Proc. of 20<sup>th</sup> IEEE Int. Symposium on Defect and Fault Tolerance in VLSI Systems, pp.371-379, 2005.
- [11] S. H. Unger, "Hazards, Critical Races, and Metastability", IEEE Transaction on Computer, June 1995, Vol. 44:6, pp. 754-768.
- [12] K. Stevens, et al., "CAD Directions High Performance Asynchronous Circuits," Proc. IEEE/ACM DAC, 1999.
- [13] M. Renaudin, "Asynchronous circuits and systems: a promising design alternative," Microelectronic Engineering, vol. 54, pp133-149, 2000.
- [14] S. M. Nowick, "Automatic Synthesis of Burst-Mode Asynchronous Controllers," PhD thesis, Stanford University, 1993.
- [15] K. Y. Yun and D. L. Dill, "Automatic synthesis of extended burst-mode circuits using generalized C-elements," in Proc. European Design Automation Conference, September 1996, pp. 290-295.
- [16] L. Davis, et al. "A data-driven machine architecture suitable for VLSI implementation," In C.L. Seitz, editor, Proc. of the Caltech Conf. on Very Large Scale Integration, pp.179-194, 1979.
- [17] Ykman-Couvreur, C.; B. Lin, "Efficient state assignment framework for asynchronous state graphs," Proc. of the Int. Conf. Computer Design, pp. 692-697, 1995.
- [18] J. Cortadella, et al. "A region-based theory for state assignment in speed-independent," IEEE Trans. on CAD, 16(8), pp. 793-812, August, 1997.
- [19] S. M. Nowick e D. L. Dill, "Exact Two-Level Minimization of Hazard-Free Logic with Multiple-Input Changes," IEEE Trans. on CAD of Integrated Circuits and Systems, Vol. 14, no 8, August 1995.
- [20] Hans Jacobson, et al. "Achieving Fast and Exact Hazard-Free Logic Minimization of Extended Burst-Mode gC Finite State machines," Proc. ICCAD, November, pp.303-310, 2000.
- [21] Michael Theobald and Steven M. Nowick, "Fast Heuristic and Exact Algorithms for Two-Level Hazard-Free Logic Minimization," IEEE Trans on CAD of Int. Circuits and Systems, vol 17, no. 11, November 1998.
- [22] K. Y. Yun, Peter A. Beerel, Vida Vakilotojar, Ayoob E. Dooply, and Julio Arceo, "The design and verification of a high-performance low-control-overhead asynchronous differential equation solver," IEEE Transactions on VLSI Systems, vol. 6, no 4, pp.643-655, Dec.1998.
- [23] R. M. Fuhrer, et al. "MINIMALIST: An Environment for the Synthesis and Verification of Burst-Mode Asynchronous Machines," in Proc. IEEE/ACM Int. Workshop Logic Synthesis, 1998.
- [24] K. Y. Yun and D. L. Dill, Automatic Synthesis of Extended Burst-Mode Circuits: Part I (Specification and Hazard-Free Implementation), Part II (Automatic Synthesis), IEEE Trans. on CAD of Integrated Circuit and Systems, Vol. 18:2, pp. 101-132, February 1999.
- [25] D. L. Oliveira, M. Strum, et al., "Miriã\_SI: a tool for the Synthesis of Speed-Independent Multi-Burst Mode Controllers," Proc. Of the 18<sup>th</sup> Symposium on Integrated Circuits and Systems, pp.56-61, 2005.