

Uma Arquitetura de Baixa Potência com Clock Pausável para MEFs Síncronas Particionadas

Duarte L. Oliveira¹, Tiago Curtinhas¹, Lester A. Faria¹, Leonardo Romano²

¹Divisão de Engenharia Eletrônica do Instituto Tecnológico de Aeronáutica
Marechal Eduardo Gomes, 50 – CEP 12.228-900 – SJG – São Paulo – Brasil

²Departamento de Engenharia Elétrica do Centro Universitário da FEI
São Bernardo do Campo – São Paulo – Brasil

Resumo — Um dos grandes desafios no projeto de sistemas digitais na atualidade está em conceber circuitos de baixo consumo potência, tendo em vista o emprego de baterias como fonte de energia e a geração de calor. Máquinas de Estado Finito (MEF) tem um importante papel na implementação destes sistemas digitais. Para solucionar este problema de consumo de potência varias técnicas são utilizadas, dentre elas está o particionamento ou decomposição de uma MEF em sub-MEFs. As sub-MEFs particionadas operam com um sinal de clock global, gerando taxas de clock sub-ótimo para as sub-MEFs menores. Este artigo propõe uma nova arquitetura para MEFs particionadas, onde cada sub-MEF tem o seu próprio clock, o que leva a cada sub-MEF a ter uma taxa de clock ótima. A nova arquitetura usa controle assíncrono para efetuar o acionamento de cada sub-MEF e este controle tem menor área, quando comparado com outros controles assíncronos da literatura.

Palavras chaves — Máquinas de estado finito, chaveamento do relógio, lógica assíncrona, particionamento, clock pausável

I. INTRODUÇÃO

O aumento da complexidade dos projetos de sistemas digitais embarcados e que são alimentados por bateria, tem no consumo de potência, o principal alvo dos projetistas, portanto estes projetos exigem um baixo consumo de potência [1]. Uma arquitetura muito utilizada para implementar um sistema digital é formada por uma rede de controladores e *datapaths* (ver Fig. 1) [2,3]. Os controladores geralmente são representados por máquinas de estado finito (MEF) e são especificados por Grafos de transições de estado (GTE) [4].

A dissipação de potência em sistemas digitais na tecnologia CMOS é composta pelas potências estática, dinâmica, de curto circuito e potência de fuga. Estudos demonstram que a maior parcela, cerca de 80%, é referente à potência dinâmica [5] e que a atividade do relógio em sistemas digitais contribui com um alto percentual no consumo de potência (15% a 45%) total do sistema [6,7].

Duarte L. Oliveira, duarte@ita.br, Tel 55+ (12) 3974-6813; Tiago Curtinhas, thiagohd@ita.br; Lester A. Faria, lester@ita.br; Leonardo Romano, leoroma@uol.com.br.

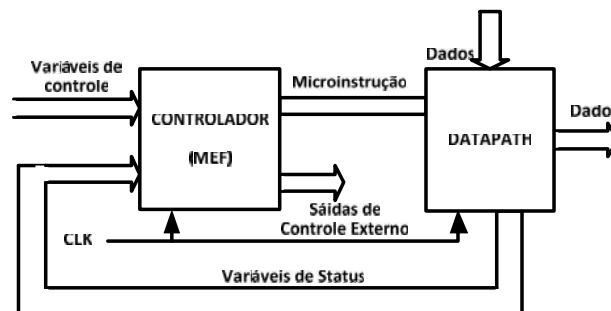


Fig. 1. Arquitetura básica de circuitos Digitais.

Objetivando a redução da potência dinâmica consumida, varias técnicas voltadas para baixa potência são utilizadas nas diversas etapas de um projeto de sistemas digitais [5,8]. Em relação às MEFs pode-se destacar o controle de inibição do sinal do relógio (*clock*) [4,9], decomposição [10-15], utilização de Flip Flop acionado nas duas bordas do sinal do clock [16,17], assinalamento de estados [18], minimização lógica e mapeamento tecnológico [19].

O método de decomposição, também conhecido por particionamento, tem demonstrado resultados significativos na redução de potência dinâmica. Ele consiste na divisão de uma MEF original ou monolítica em N sub-MEFs. Quando uma das sub-MEF está operando, as outras ficam em estado de repouso (não consumindo potência). Assim que o ciclo de operações termina outra sub-MEF é acionada, até que todas tenham sua operação realizada.

Um circuito de controle do sinal do *clock* geralmente chamado de Bloco de Controle do *Clock* (BCC) é utilizado para efetuar a comutação entre as sub-MEFs, permitindo que uma única sub-MEF opere de cada vez, promovendo, assim a redução da potência dinâmica consumida. Diferentes BCCs foram propostos, dentre os quais as propostas interessantes de [4] [20-22]. Podemos classificar os BCCs em dois modos de atuação, que são: BCC síncrono [4] e BCC assíncrono [20-22]. Estudos realizados por Oelmann et al. [20] mostram que a versão assíncrona traz importante redução no consumo de potência em relação à versão síncrona, mas tem como desvantagem um aumento na área utilizada pelo circuito e um maior tempo de ciclo. Os métodos de particionamento de MEFs que adotam o controle síncrono são [10-12] e que adotam o controle assíncrono são [13-16] [23].

Sistemas digitais de alta integração na tecnologia DSM-MOS (*Deep-Sub-Microm - MOS*) operam com múltiplos clocks e não com um simples *clock* global. Portanto, um subsistema pode trabalhar com diferentes *clocks* ou fases diferentes ou até usando *clock* local. Em um sistema digital heterogêneo um módulo pode operar ou com *clock* local, ou com *clock* externo. Das diferentes arquiteturas propostas para MEF particionadas, todas têm em comum o uso do *clock* externo, operando como *clock* global. Como o *clock* global é definido pela sub-MEF com o pior caminho crítico, portanto para as outras sub-MEFs a taxa de *clock* é sub-ótima. Uma decomposição muito comum nas propostas de particionamento é ter sub-MEFs grandes, mas com probabilidade de pouco processamento e sub-MEFs pequenas, mas com alta probabilidade de processamento. Esta estratégia que permite uma maior redução de potência tem como limitação o uso de uma taxa de ciclo média sub-ótima.

Neste artigo propomos uma arquitetura voltada para MEFs particionadas, onde cada sub-MEFs tem o seu próprio *clock*, portanto cada sub-MEF tem a sua própria taxa de *clock* (ver Fig. 2). A nova arquitetura permite que todas as sub-MEFs operem com a taxa de *clock* ótima. Este *clock* pode ser local ou externo. Neste artigo propomos o uso do gerador de *clock* pausável (com pausa) que é robusto à metaestabilidade. A ativação e a desativação das sub-MEFs na nossa arquitetura é realizada por um controle assíncrono que é especificado no modo *burst* [24]. A especificação do controle está relacionada com a interação entre as sub-MEFs.

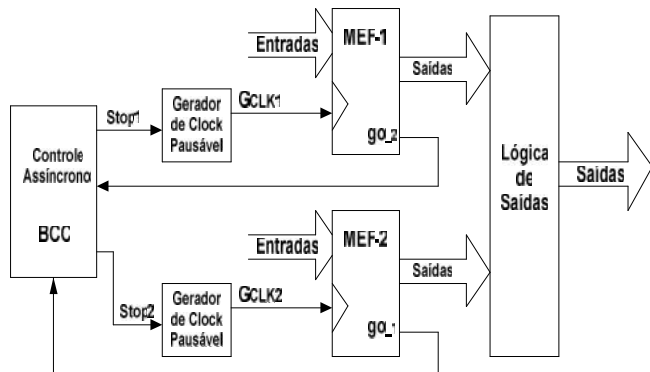


Fig. 2. Arquitetura com gerador de *clock* pausável proposta.

II. TRABALHOS RELACIONADOS

A. Circuito de controle síncrono

Benini et al. [4] propôs uma comunicação síncrona entre as sub-MEFs. Um sinal de relógio global realiza a comutação entre as sub-MEF particionadas, portanto há um sincronismo entre essas sub-máquinas. Fig. 3 apresenta a versão síncrona proposta em [4], onde o circuito de comunicação entre as sub-MEF é ativado com o sinal do relógio. Os blocos Fa_1 e Fa-2 são circuitos combinatórios e devem ser projetados em função da interação das sub_MEFs particionadas.

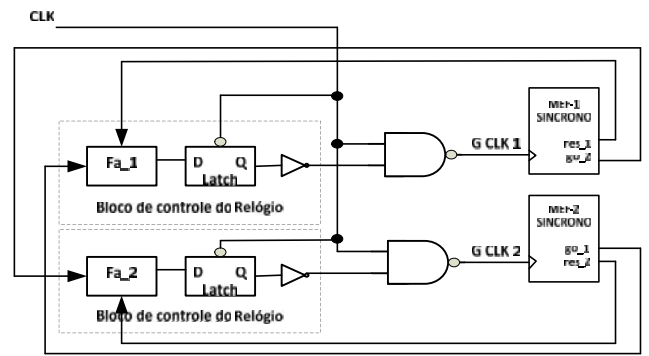


Fig. 3. Arquitetura síncrona proposta em [4].

B. Circuito de controle assíncrono

Oelmann et al. em [20] propôs o circuito de controle do relógio na versão assíncrona, proporcionando uma redução no consumo de potência em relação a versão síncrona. Como limitação há um aumento de área e o tempo de ciclo do controle pode afetar a taxa do relógio. Fig. 4 apresenta a versão assíncrona proposta em [20], onde o circuito de controle entre as sub-MEF não utiliza o sinal do relógio para acionar o circuito de controle do relógio.

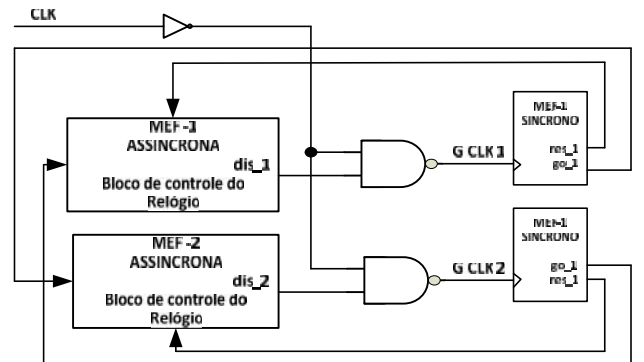


Fig. 4. Arquitetura assíncrona proposta por [20].

Fig. 5 apresenta o circuito do BCC na versão assíncrona [20]. O controle possui dois sinais de entrada *res_1* e *go_2* e um sinal de saída *disp_1*. Este circuito deve ser utilizado para o controle de cada sub-MEF particionada.

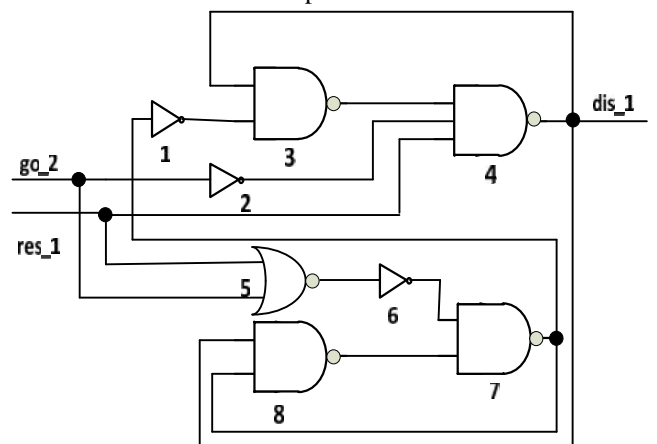


Fig. 5. Circuito assíncrono de controle do relógio.

A primeira vantagem do circuito de controle assíncrono em relação ao circuito de controle síncrono é que esta célula proposta é fixa, ou seja, não depende da interação entre as sub-MEF, enquanto na configuração síncrona o circuito deve ser projetado em função da interação entre as sub-MEF. Ambas as propostas são utilizadas nas arquiteturas onde as MEF particionadas possuem dois sinais de controle, sendo que um sinal é utilizado para acionar o circuito de controle do clock da outra sub-MEF e outro sinal desativa o sinal do clock da sub-MEF em análise.

C. Circuito de controle assíncrono modo-burst

Oliveira et al. [21] propôs uma variante do BCC assíncrono que usa BCCs fixos [20]. Na proposta em [21] a especificação do BCC está relacionado com interação entre as sub-MEFs, o que acarreta como vantagem uma redução de área e de potência e como desvantagem um custo maior de projeto [22]. Esta vantagem é obtida, porque o BCC somente necessita de um único sinal de saída, para cada sub-MEF (ver Fig. 6). Como ilustração, Fig. 7 mostra o GTE do benchmark DK25 particionado em duas partições pela ferramenta LSPart [23]. Fig. 8 mostra a especificação modo *burst* do controle e Fig. 9 mostra o circuito lógico sintetizado pela ferramenta Minimalist [25], onde a variável de estado (*sv*) foi introduzida para eliminar os conflitos.

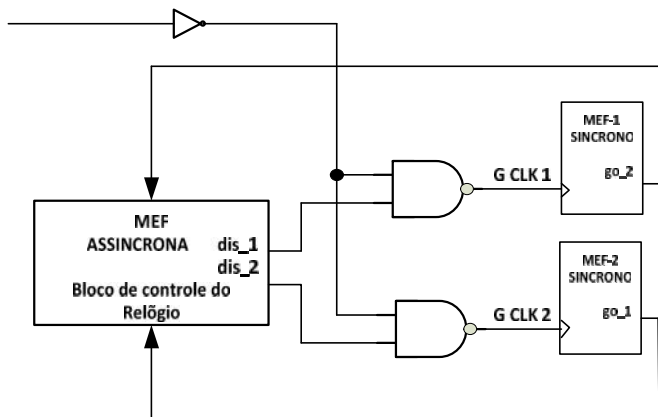


Fig. 6. Arquitetura assíncrona proposta por [21].

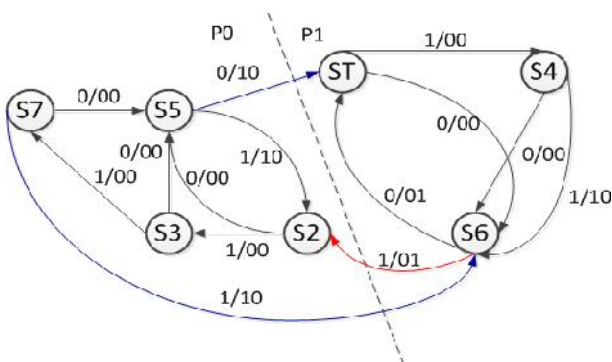


Fig. 7. MEF DK25 particionada em 2 sub-MEFs [22,23].

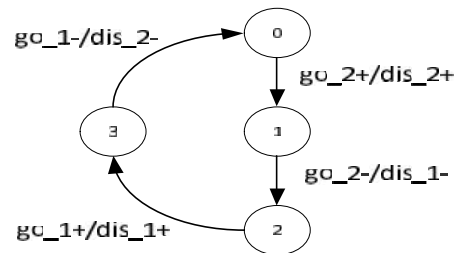


Fig. 8. Especificação modo *burst*: controle para 2 sub-MEFs do DK25 [22,23].

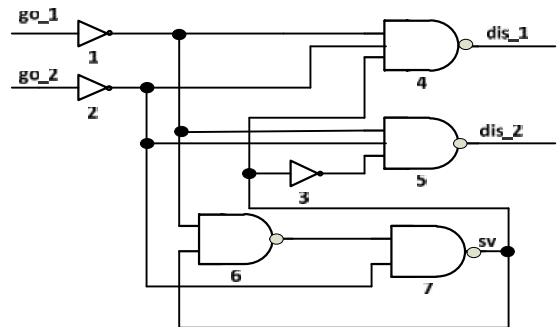


Fig. 9. Circuito lógico do BCR para 2 sub-MEF.

D. Método de particionamento

O método de particionamento adotado neste artigo foi proposto em [23]. Ele insere dois estados em cada partição e um sinal que indica qual sub-MEF vai ser ativada. Fig. 10 mostra um particionamento que acarreta em duas partições e Fig. 11 mostra as duas sub-MEFs resultantes do particionamento com a introdução de dois estados e do sinal *go_x*.

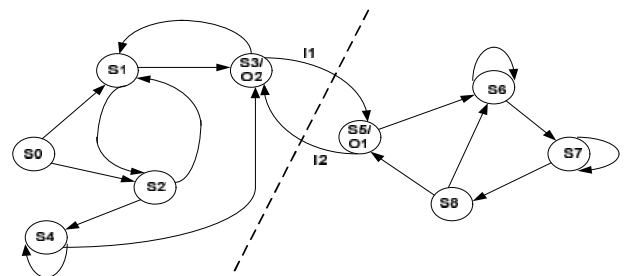


Fig. 10. GTE particionadas em duas sub-MEFs [23].

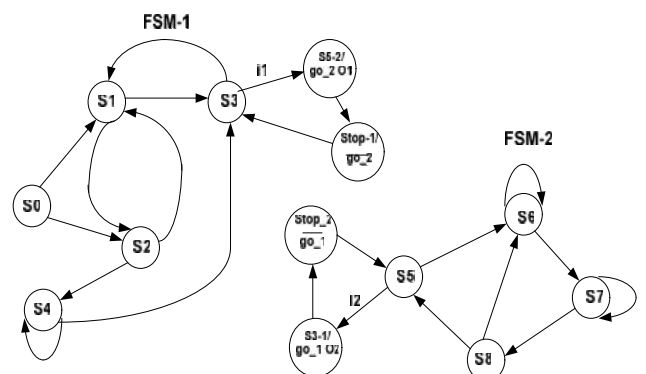


Fig. 11. GTE transformada em duas sub-GTEs de [23].

III. ARQUITETURA COM *CLOCK PAUSÁVEL* PROPOSTA

Este artigo propõe uma nova arquitetura para MEFs síncronas particionadas. Diferente das arquiteturas propostas na literatura que usam um *clock* global para todas as sub-MEFs, a nova arquitetura tem como característica o uso de múltiplos *clocks*, isto é cada sub-MEF tem o seu próprio *clock*, que é gerado localmente. Fig. 2 mostra a arquitetura proposta com duas sub-MEFs, onde as duas sub-máquinas possuem os seus geradores de *clock* pausável (GCP). Os GCPs são controlados por um controle assíncrono, que é especificado no modo *burst* e que descreve a interação entre as sub-MEFs. O controle assíncrono proposto usa somente um único sinal (*go_x*) para definir qual a sub-MEF que será ativada. Fig. 12 mostra o GCP proposto em [26] e usado em nossa arquitetura.

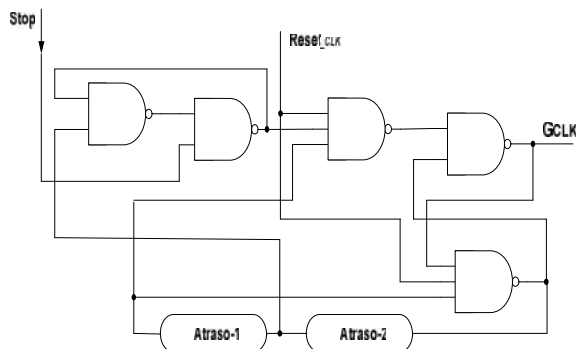


Fig. 12. Gerador de clock pausável de [26].

IV. METODOLOGIA PARA SÍNTESE DO BCC ASSÍNCRONO VOLTADA PARA ARQUITETURA DE *CLOCK PAUSÁVEL*

A síntese do BCC assíncrono voltada para a arquitetura de *clock* pausável tem quatro passos e é baseada em [21,22]:

- 1) Capturar a descrição da MEF a ser particionada.
- 2) Determinar o número de partições através de um espaço de projeto e obter as sub-MEFs utilizando a ferramenta de particionamento, LSPart [23].
- 3) Obter as interações entre as sub-MEFs e descrever o BCC assíncrono utilizando a especificação modo *burst* [24].
- 4) Utilizar a ferramenta SAGAAs [27] sintetizar o BCC assíncrono.

Para demonstrar a metodologia da síntese do BCC proposto será utilizada a MEF DK25, que foi particionada em 2, 3 e 4 partições pela Ferramenta LSPart.

A. Duas partições

Fig. 7 mostra a MEF DK25 com a identificação de duas partições. Fig. 13 mostra a especificação modo *burst* e Fig. 14 mostra o circuito lógico que foi sintetizado na ferramenta SAGAAs que controla duas sub-MEFs.

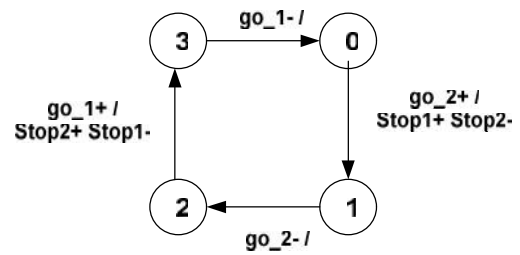


Fig. 13. Especificação modo *burst* do BCC assíncrono

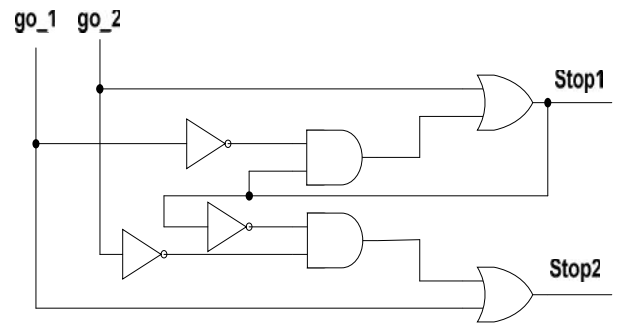


Fig. 14. Circuito lógico do controle: BCC

B. Estudo de caso: três partições

A Fig. 15 apresenta o GTE da MEF DK25 com a identificação de três partições.

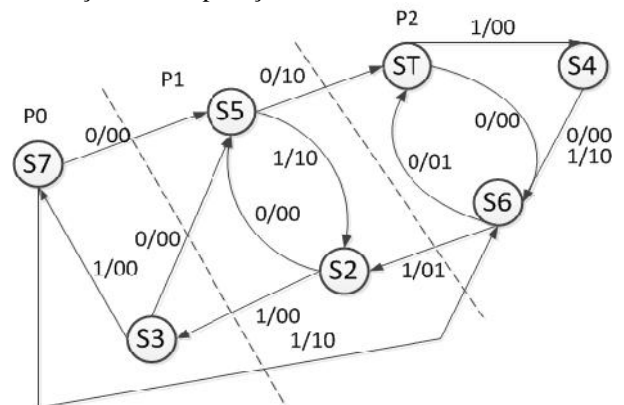


Fig. 15. MEF DK25 particionada em 3 sub-MEF [22].

A Fig. 16 apresenta a especificação modo *burst* do BCC assíncrono que controla as três sub-MEFs.

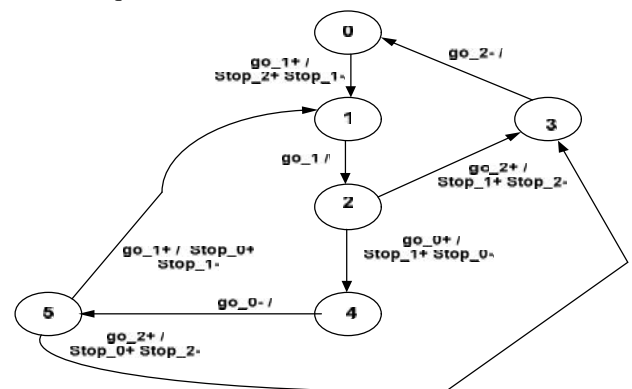


Fig. 16. Especificação modo *burst* do BCC.

C. Estudo de caso: quatro partições

A Fig. 17 apresenta GTE da MEF DK15 com a identificação de quatro partições.

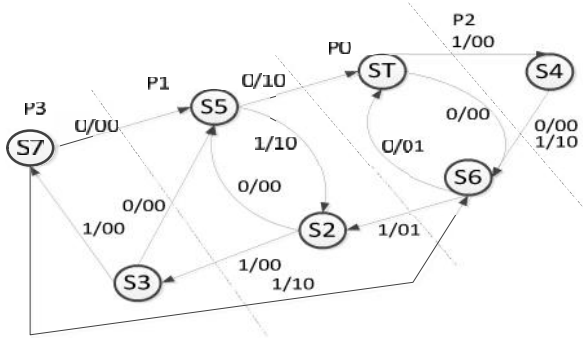


Fig. 17. MEF DK25 particionada em 4 sub-MEF [22].

A Fig. 18 apresenta a especificação modo *burst* do BCC assíncrono que controla quatro sub-MEFs.

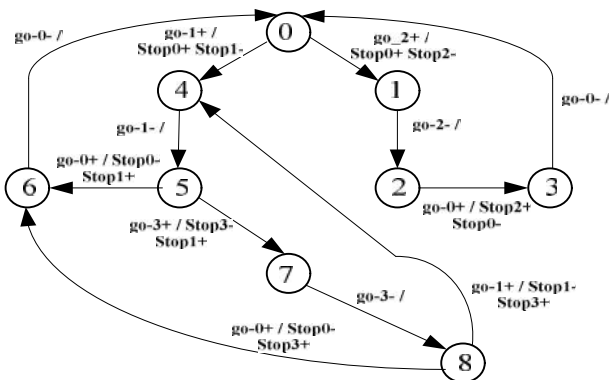


Fig. 18. Especificação modo *burst* do BCC.

V. SIMULAÇÃO & RESULTADOS

A arquitetura proposta é simulada para duas sub-MEFs na ferramenta ALTERA QUARTUS II [28]. Fig. 19 mostra o diagrama de simulação da arquitetura proposta e que satisfaz a especificação do controle. O circuito mostrou ser livre de *hazard* conforme o esperado.

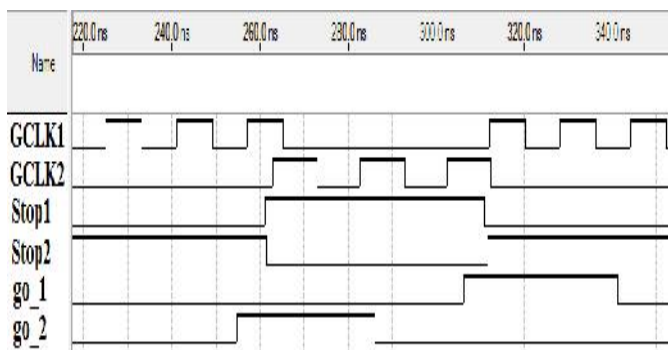


Fig. 19. Simulação da arquitetura com clock pausável para duas sub-MEFs.

Nesta seção vamos fazer uma comparação de área dos BCCs de [20] e [21,22] e com a nossa proposta. Esta comparação é realizada para 2, 3, e 4 partições, que envolve o benchmark DK25 e que foi o nosso estudo de caso. Os BCCs obtidos são livres de *hazard*.

A tabela I mostra que o nosso BCC obteve uma redução de 66% no número de produtos; uma redução de 63% no número de literais e uma redução de 53% no número de transistores, quando comparado com [20]. Quando comparado com [22] obteve uma penalidade de 50% no número de produtos e uma redução de 20% no número de literais.

A tabela II mostra que o nosso BCC obteve uma redução de 66% no número de produtos; uma redução de 72% no número de literais e uma redução de 50% no número de transistores, quando comparado com [20]. Quando comparado com [22] obteve uma redução de 40% no número de produtos; uma redução de 50% no número de literais e uma redução de 38% no número de transistores.

A tabela III mostra que o nosso BCC obteve uma redução de 66% no número de produtos; uma redução de 72% no número de literais e uma redução de 41% no número de transistores, quando comparado com [20]. Quando comparado com [22] obteve uma redução de 20% no número de produtos; uma redução de 33% no número de literais e uma redução de 33% no número de transistores.

TABELA I RESULTADOS – 2P

MEF em duas partições: DK25				
Controle	Especificação N_en/N_s/N_es/N_t	Nrc. Produtos	Nrc. Literais	Nrc. Transistores
BCC de [20]	Duas BCCs	6	22	56
BCC de [22]	2/2/4/4	1	10	26
BCC Proposta	2/2/4/4	2	8	26

TABELA II RESULTADOS – 3P

MEF em três partições: DK25

Controle	Especificação N _{en} /N _s /N _{es} /N _t	Nro. Produtos	Nro. Literais	Nro. Transistores
BCC de [20]	Três BCCs	9	33	84
BCC de [22]	5/2/8/10	5	18	68
BCC Proposta	3/3/6/8	3	9	42

TABELA III RESULTADOS – 4P

MEF em quatro partições: DK25

Controle	Especificação N _{en} /N _s /N _{es} /N _t	Nro. Produtos	Nro. Literais	Nro. Transistores
BCC de [20]	Quatro BCCs	12	44	102
BCC de [22]	5/2/8/10	5	18	90
BCC Proposta	3/3/6/8	4	12	60

VI. CONCLUSÃO

Particionar uma MEF em sub-MEFs é interessante para reduzir a complexidade da síntese, principalmente para MEF de grande porte. Um interesse maior do particionamento em sub-MEFs está no grande potencial de economizar energia. O motivo é que enquanto uma sub-MEF está processando, as outras estão em repouso, isto é, sem consumir energia. Um controle eficiente para ativar e desativar sub-MEFs é o controle assíncrono. As arquiteturas propostas para MEFs particionadas envolve um sinal de *clock* global, o que leva sub-MEFs a operar com taxas de *clock* sub-ótimo. Este artigo propõe uma nova arquitetura para MEFs particionadas, onde cada sub-MEF opera com a taxa de *clock* ótima. Para esta arquitetura um novo controle assíncrono é proposto. Ele leva a uma redução de área e potencialmente a uma redução de potência dissipada, quando comparado com outras propostas para controle do *clock*. Como trabalho futuro, pretendemos realizar uma avaliação envolvendo: área, potência dissipada e tempo de ciclo da nova arquitetura. Também se pretende aplicar a nossa arquitetura em sistemas embarcados baseados em rede de controladores e implementar em dispositivos programáveis.

REFERENCIAS

[1] C. Piguet, "Low-Power CMOS Circuits Technology, Logic Design and CAD Tools," Taylor & Francis Group, 2006.
[2] H. Hsieh, F. Balarin et. al. "Synchronous approach to the FunctionalEquivalence of Embedded System Implementations," IEEE

Trans. On CAD of Int. Circuits and Systems, vol.20, no.8, pp.1016-1033, August 2001.
[3] J. J. Rodriguez, et. al., "Features, Design Tools, and Applications Domains of FPGAs," IEEE Trans. On Industrial Electronics, vol. 54, no 4, pp. 1810-1823, August, 2007.
[4] L. Benini and G. De Micheli, "Automatic Synthesis of Low-Power Gated-Clock Finite-State Machines," IEEE Trans. on CAD of Integrated Circuits and Systems, Vol.15, No.6, pp.630-643, June 1996.
[5] L. Benini and G. De Micheli, "Systems-Level Power Optimization: techniques and Tools," ACM Trans. on Design Automation of Electronic Systems, 5(2), pp. 115-192, April 2000.
[6] A. Jain et al., "A 1.2 GHz alpha microprocessor with 44.8 GB/s chip pin bandwidth," in IEEE Int. Solid-State Circuits Conf. Tech. Dig., pp. 240-241, February 2001.
[7] F. N. Najm, "A Survey of Power Estimation Techniques in VLSI Circuits," IEEE Trans. on VLSI Systems, vol. 2, no. 4, pp.446-455, December 1994.
[8] S. Devadas and S. Malik, "A Survey of Optimization Techniques Targeting Low Power VLSI Circuits", Proc. 32nd ACM/IEEE DAC, pp.242-247, 1995.
[9] Q. Wu, M. Pedram, and X. Wu, "Clock-Gating and Its Application to Low Power Design of Sequential Circuits," IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications, vol. 47, no.103, pp.415-420, March 2001.
[10] L. Benini and G. De Micheli, "Synthesis of Low-Power Selectively-Clocked Systems from High-Level Specification," ACM Trans. on Design Automation of Electronic System, 5(3), pp. 311-321, July 2000.
[11] J. C. Monteiro and A. L. Oliveira, "Implicit FSM Decomposition Applied to Low-Power Design," IEEE Trans. on VLSI Systems, Vol.10, No. 5, pp.560-565, October 2002.
[12] B. Liu, et al., "FSM Decomposition for Power Gating Design Automation in Sequential Circuits", 76th Int. Conf. on ASIC, ASICON, pp.944-947, 2005.
[13] B. Oelmann, et al., "Automatic FSM Synthesis for Low-Power Mixed Synchronous/Asynchronous Implementation," Journal of VLSI Design, Special Issue on Low-Power Design, vol. 12, no.2, pp. 167-186, 2001.
[14] C. Cao, et al., "Synthesis Tool for Low-Power Finite-State Machines with Mixed Synchronous/Asynchronous State Memory," IEE Proc. Comput. Digit. Tech. vol. 153, no. 4, pp.243-248, July 2006.
[15] C. Cao and B. Oelmann, "Low-Power State Encoding for Partitioned FSMs with Mixed Synchronous/Asynchronous State Memory," Integration the VLSI Journal, vol. 41, pp.123-134, 2008.
[16] P. Zhao, J. McNeely, et al., "Low-Power Clock Branch Sharing Double-Edge Triggered Flip-Flops," IEEE Trans. on VLSI Systems, vol. 15, no.3, pp.338-345, March 2007.
[17] D. L. Oliveira, L. S. Ferreira, et al "Synthesis of Synchronous Digital Systems Operating in Double-Edge of Clock," LASCAS 2012 3rd IEEE Latin American Symposium on Circuits and Systems, Playa del Carmen, México.
[18] P. Baccheletta et al. "Low-Power State assignment Techniques for Finite State Machines," IEEE Int. Symposium on Circuits and systems, Geneva, Switzerland, pp.641-644, May 2000.
[19] J.-Mou Tseng and J.-Yang Jou, "A Power-Driven Two-Level Logic Optimizer," Proc. Of the ASP-DAC, pp.113-116, 1997.
[20] B. Oelmann, et al., "Asynchronous Control of Low-Power Gated-Clock Finite State Machines," Proc. IEEE Int. Conf. on Electronics, Circuits and Systems, pp. 915-918, 1999. .
[21] D. L. Oliveira and L. S. Ferreira, "Asynchronous Burst-Mode Control for Low-Power Gated-clock Finite State Machines," 2nd IEEE Latin American Symposium on Circuits and Systems, LASCAS 2011. Bogotá, Colômbia.
[22] L. S. Ferreira, "Controle Assíncrono para Acionamento de Máquinas de Estados Finito Particionadas," XV Simpósio de aplicações operacionais em área de defesa (SIGE), 2013.
[23] L. S. Ferreira, "Particionamento de Máquinas de Estado Finito Síncronas com Controle Assíncrono Visando Redução do Consumo de Potência," Tese de Mestrado, ITA 2012.
[24] S. M. Nowick, "Automatic synthesis of burst-mode asynchronous controllers," Ph.D. Thesis, Technical report CSL-TR-95-686, 1995.

- [25] R. M Fuhrer et al., "Minimalist: An environment for the Synthesis, verification and traceability of burst-mode machines," Technical Report, Columbia University, TR-CUCS-020-99, 1999.
- [26] D. L. Oliveira, et al., "Design of an Improved and Robust Asynchronous Wrapper (AW) for FPGA Applications," Journal of Integrated Circuits and Systems, vol. 8, nro.1, pp.54-63, 2013.
- [27] T. Curtinhas, et al., "A novel state assignment method for Extended Burst-Mode FSM design using Genetic Algorithm," 27th Symposium on Integrated Circuits and Systems Design," paper accepted, 2014.
- [28] ALTERA Corporation-www.altera.com, acesso em 2014.