

POR QUE UM CIRCUITO COMPUTACIONAL PODE TRABALHAR SEM RELÓGIO ?¹

*Leo Weber*²

Resumo: O propósito deste artigo é fazer uma breve introdução aos conceitos relacionados à lógica assíncrona e mostrar características operacionais significativas de sua aplicabilidade contemporânea, citando alguns exemplos de recentes desenvolvimentos e pesquisas na área.

Palavras chave - Arquitetura de computadores. Lógica assíncrona. Sistemas digitais sem relógio.

Abstract: The purpose of this article is to do an introduction about concepts related with asynchronous logic and to do an overview of the important operational features in contemporary applications, with examples of the developments and researches in the area.

Key words - Computer's architecture. Asynchronous logic. Digital systems without clock.

1. Uma breve retrospectiva histórica

A história dos computadores começou no momento em que o homem sentiu a necessidade de realizar cálculos complexos com maior precisão e rapidez. A idéia de que a informação podia ser codificada, para em seguida ser processada independentemente do sentido das mensagens, foi se formando aos poucos: estava aberto o caminho para que o cálculo se tornasse progressivamente uma questão de máquinas e não mais de instrumentos. [BRET]

A transformação de uma determinada informação de entrada, em outra de saída, contempla o conceito de processamento. Nele reside a generalidade da estrutura de um sistema digital e também a diferença entre processamentos que consomem muita energia (processamento industrial, por exemplo) ou envolvem pequena quantidade de energia (processamento computacional, por exemplo). O cálculo é o processamento por excelência da informação [LEVY].

Os primeiros computadores eram compostos exclusivamente por elementos mecânicos. Eles herdaram força nos automatismos, tradição ancorada nas técnicas e na busca da construção de seres artificiais. Os primeiros autômatos eram “engenhocas” mecânicas que reproduziam partes humanas ou seu todo, segundo [LOSA], para, em seguida, derivar para aplicações no controle de processos e na regulação de sistemas realimentados. Estava aí posta a necessidade crescente da realização de cálculos, inicialmente muito simples. À medida que os processos tornaram-se mais complexos, também a carência de executar operações nas mesmas dimensões aumentou. Pascal, em 1642, desenvolveu uma máquina de calcular totalmente mecânica e, no século XIX, Babbage construiu a máquina de diferenças, capaz de calcular tabelas matemáticas. Mas foi Hollerith, no final do século XIX, quem concebeu a idéia de processar dados a partir de cartões perfurados, importante marco na técnica de processamento de dados.

Desde os primeiros projetos implementados, inicialmente com a utilização de relés e válvulas eletrônicas, que os computadores conheceram aperfeiçoamentos incessantes no que diz respeito a seus componentes eletrônicos (hardware), chegando até os modernos circuitos integrados. No entanto, os princípios de base e a forma como os elementos de um computador estão organizados,

¹ Pesquisa suportada pela Fapergs (Fundação de Amparo a Pesquisa do Estado do RS)

² Professor da Fundação Liberato Salzano e da Ulbra (Universidade Luterana do Brasil)

sua arquitetura, continuam os mesmos [BRET]. A razão disto está no projeto desenvolvido na década de 40 por Von Neumann, matemático que concebeu a organização das máquinas computacionais. Mesmo com o advento dos microprocessadores, eles continuam sendo compostos por dispositivos de entrada, de saída, unidade de memória, unidade aritmética e unidade de controle, com processamento serial, sincronismo dos eventos a partir de um oscilador (relógio), controle centralizado e comunicação baseada em endereçamentos [Figura 1]. O fluxo de dados entre o processador e a memória é o gargalo de um computador sequencial comandado por relógio. A origem do problema é que a memória foi projetada para acessar uma única localização em cada ciclo de instrução. Enquanto mantivermos este projeto, a maneira de aumentar a velocidade do computador será reduzindo o tempo do ciclo, ou seja, aumentando a frequência do relógio [HILL]. Outra técnica seria a utilização do processamento paralelo, com vários processadores operando simultaneamente.

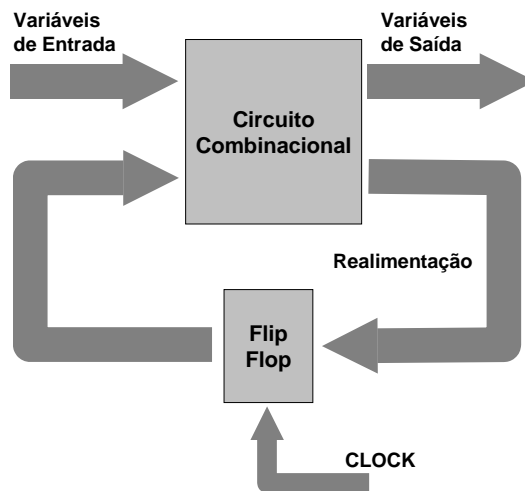


Figura 1: Máquina síncrona

2. O que é lógica assíncrona e como este conceito se aplica a sistemas computacionais

O relógio é implementado por um cristal oscilador que sincroniza todo o funcionamento de um computador, controlando o tempo de cada um dos eventos em cada um dos circuitos de todos os chips que compõe o sistema. Em oposição a esta concepção, um sistema assíncrono, ou sem relógio, é baseado na operação por eventos, fazendo com que diferentes bits de um chip operem com velocidades diversas, enviando e recebendo dados quando for apropriado. A este tipo de arquitetura computacional dá-se o nome de alternativa sem relógio, ou assíncrona, porque nela o funcionamento dos circuitos não depende da tirania de um relógio [Figura 2] [MART].

A idéia de projetar computadores digitais com lógica assíncrona esteve presente desde os primeiros sistemas. O Modelo Geral de um Sistema Digital, proposto por Phister, relaciona saídas e entradas por meio de um sistema simultâneo de funções digitais [PHIS]. Mas o desenvolvimento baseado em relógio predominou até hoje, porque é mais fácil projetar chips nos quais somente acontecem coisas quando o relógio define. Mais recentemente, os chips ficaram maiores e mais complexos e a necessidade de distribuição dos sinais do relógio em todo o circuito integrado tornou-se uma tarefa nada fácil. Outro problema a ser considerado é a perda de energia proporcionada pelas partes inativas do circuito, que tem que responder a cada ciclo de relógio. Nota-se que, hoje em dia, em média, 15% da circuitaria de um chip é projetada para distribuir o sinal de relógio, e mais de 20% da potência é consumida pelo próprio relógio [MARK].

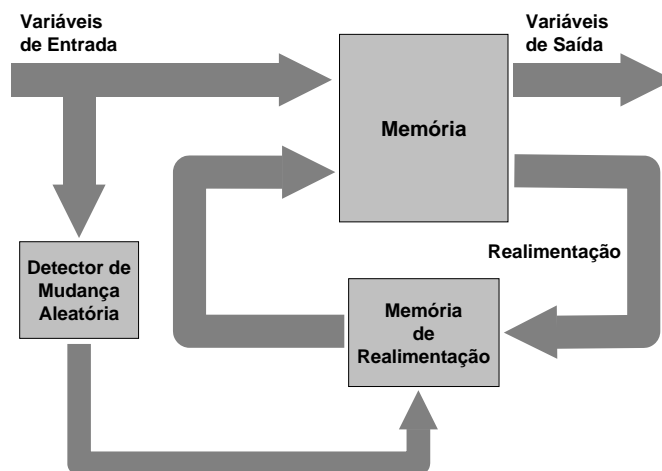


Figura 2: Máquina assíncrona

3. Quais as vantagens de um sistema computacional assíncrono ?

Há várias concepções de projeto com lógica assíncrona, nas quais grupos de pesquisa atuam simultaneamente. Um primeiro exemplo é o *Amulet Group - University of Manchester*, na Grã-Bretanha, que desenvolveu uma arquitetura comercial do microprocessador *ARM* baseado em lógica assíncrona. Os microprocessadores *Amulet 1, 2 e 3i* possuem várias aplicações com excelente performance e baixo consumo de energia. Outro exemplo é o *Asynchronous Design Group - Sun Microsystems Laboratories*, nos Estados Unidos. Recentes demonstrações apresentaram chips projetados com chaveamento não comandados por relógio, como, por exemplo, os apresentados em [SUTH].

Várias vantagens podem ser arroladas para justificar o desenvolvimento de sistemas computacionais sem relógio [EBER], quais sejam:

3.1. Velocidade:

Pesquisas demonstram que será possível dobrar a velocidade de chaveamento de circuitos assíncronos em relação a circuitos convencionais baseados em relógio, o que significa, sem dúvida, um grande diferencial. Por se tratar de um sistema baseado em eventos, não há relógio e, portanto, quem limita a velocidade é o tempo necessário para propagar a informação por seus circuitos, que é a velocidade da luz. Para exemplificar, e por curiosidade, a luz viaja acerca de 30 cm/ns. O tempo de ciclo dos mais rápidos computadores da atualidade é de cerca de 1 ns, e não é coincidência que seus processadores tenham menos de 30 cm de um lado a outro. Estamos chegando ao limite de quanto poderemos acelerar o computador sem alterar seu projeto básico [HILL].

3.2. Modularidade:

A reutilização da circuitaria é outro aspecto essencial, pois assim pode-se buscar o aumento da velocidade de processamento sem que isto signifique obsolescência, seja ela programada ou não. Para constar, só no ano de 2000, 33 milhões de máquinas computacionais ficaram desatualizadas nos EUA, das quais 6 em cada 10 viraram lixo eletrônico, sucata descartável. No Brasil, em 1999, mais de 500.000 PCs chegaram aos lares brasileiros, dobrando este número em 2000, levando à aposentadoria 1 velho PC para cada novo equipamento adquirido [VEJA].

3.3. Menor consumo de energia:

O ambiente baseado em transições por eventos não apresenta a necessidade de distribuição do sinal do relógio, o que é extremamente complexo nos chips atuais, nem apresenta "jitter", que são as variações indesejáveis do relógio. Disto advém que o consumo de energia numa arquitetura com relógio é maior, porque mesmo as partes inativas dos circuitos têm que responder a cada ciclo do mesmo, o que não ocorre em sistemas assíncronos.

3.4. Menor emissão eletromagnética:

Os chips com relógio produzem emissões eletromagnéticas em frequências múltiplas do oscilador (harmônicas), que podem causar rádio interferência (RFI). Um belo exemplo disto é a comparação apresentada por [NOWI] entre o espectro de frequências medido na fonte de alimentação de uma versão síncrona e de uma versão assíncrona do microcontrolador 80C51 [Figura 3].

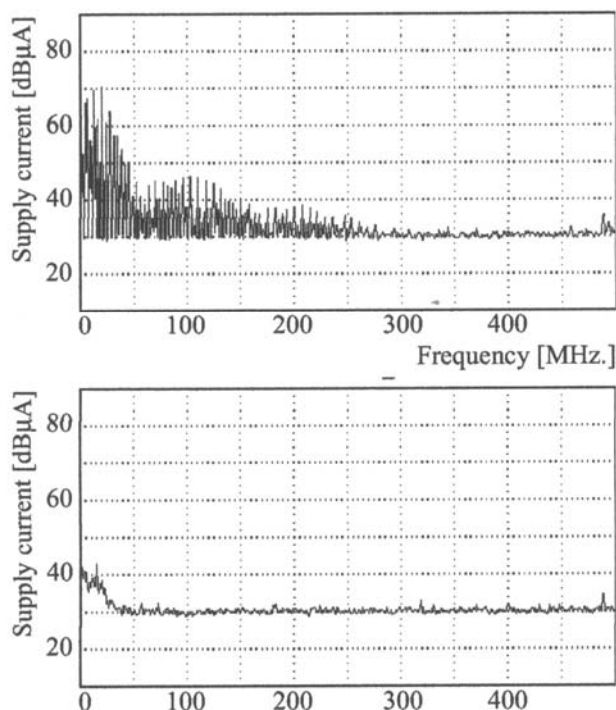


Figura 3: Espectro de frequências da corrente de alimentação medida em uma versão síncrona (em cima) e em uma versão assíncrona (embaixo) do microcontrolador 80C51. Os dois microcontroladores usaram a mesma biblioteca, foram implementados em processo CMOS de $0,5 \mu$ e rodaram o mesmo programa de teste com a mesma performance (Gráficos fornecidos por Philips Semiconductors Zürich) [NOWI]

4. As possibilidades futuras:

Hoje há ainda um grupo reduzido de pesquisadores que acreditam nas potencialidades das aplicações de circuitos computacionais baseados em lógica assíncrona. Isto se dá fundamentalmente pela descrença de que os projetistas da indústria de chips venham a alterar o caminho adotado até agora. Os argumentos são fortes. Diz a maioria incrédula que para cada oportunidade proporcionada

pela lógica assíncrona existe uma alternativa síncrona compatível. Mas isto tem servido para gerar estudos cada vez mais consistentes sobre a lógica assíncrona.

As soluções que adotam lógica assíncrona tendem a ser cada vez mais efetivas, baratas e de fácil implementação. Um bom exemplo é a vantagem explícita de menor emissão eletromagnética, como o demonstrado pela Philips em relação ao microcontrolador 80C51. Em outros casos, são as aplicações que mesclam circuitos síncronos e assíncronos, com sub-circuitos dedicados e integrados, como os da Myricon, uma pequena empresa de alta tecnologia do Vale do Silício, EUA, que desenvolve sistemas de interconexão para a IBM [MARK]. Mais um exemplo é o processador de origem inglesa ARM, utilizado em computadores de mão e telefones sem fio. Ou ainda, o caso da Philips que tem um "pager" baseado em lógica assíncrona [ECON]. Na verdade, soluções como as citadas só serão utilizadas se apresentarem claras e definitivas vantagens em relação aos sistemas síncronos, ou quando não houver alternativas economicamente viáveis de concepção convencional.

5. Bibliografia

[BRET] BRETON, Philippe. **História da Informática**. São Paulo: UNESP, 1991.

[EBER] EBERGEN, Jo. **Circuits Without Clocks: What Makes Them Tick?** Mountain View, Sun Microsystems Laboratories, May 13th, 1998.

[ECON] OLD tricks for new chips. **The Economist**, London, April 19th, 2001. Science and Technology.

[HILL] HILLIS, Daniel. **O Padrão Gravado na Pedra**. Rio de Janeiro: Rocco, 2000.

[LEVY] LÉVY, Pierre. **A Máquina Universo**. Porto Alegre: Artmed, 1998.

[LOSA] LOSANO, Mario G. **Histórias de Autômatos**. São Paulo: Companhia das Letras, 1992.

[MARK] MARKOFF, John. Computing Pioneer Challenges the Clock. **New York Times**, New York, March 5th, 2001.

[MART] MARTINS, W. Waneck. **Esção (n-m-p): Um Computador não-Von Neumann**. São Paulo: Cartgraf, 1985.

[NOWI] VAN BERKEL, C. H.; JOSEPHS, Mark B.; NOWICK, Steven M. Scanning the Technology: Applications of Asynchronous Circuits. **Proceedings of the IEEE**, v.87, n.2, p.223-233, Feb. 1999.

[PHIS] PHISTER Jr., M. **Logical Design of Digital Computers**. New York: John Wiley & Sons, 1958.

[SUTH] SUTHERLAND, Ivan E.; LEXAU, Jon K. Designing Fast Asynchronous Circuits. **ASYNCR 2001 Seventh International Symposium on Advanced Research in Asynchronous Circuits and Systems**, Salt Lake City, Proceedings, March 11-14, 2001.

[VEJA] ENTULHO Digital. **Veja**, São Paulo, p.105, 7 de junho de 2000. Computador.