

# Processadores Low-Power/Energy Efficient

Alexander Peter Sieh, 087984

Centro de Pesquisas Wernher Von Braun  
Alice de Castro P. N. Mattosinho, 301  
Alphaville, 13098-392 Campinas-SP, Brasil  
Fone/Fax: +55(19) 3262-2207  
alexsieh@gmail.com

José Arnaldo Bianco Filho, 019222

Centro de Pesquisas Wernher Von Braun  
Alice de Castro P. N. Mattosinho, 301  
Alphaville, 13098-392 Campinas-SP, Brasil  
Fone/Fax: +55(19) 3262-2207  
josearn@gmail.com

## RESUMO

Este artigo resume técnicas para atingir baixa potência e eficiência energética no desenvolvimento de processadores.

### Termos Gerais

Desempenho, Teoria.

### Palavras-chave

Processador, pipeline, potência dinâmica, potência estática, técnicas low-power, energy-efficient, gating.

## 1. INTRODUÇÃO

Devido às atuais necessidades de mercado, consumo e eficiência energética em processadores estão se tornando cada vez mais relevantes. Neste contexto, técnicas tanto de arquitetura quanto de design são extensivamente empregadas a fim de se conquistar os melhores resultados para cada tipo de utilização. As técnicas aqui descritas focam o desenvolvimento em ASIC, mas também contemplam técnicas relacionadas à arquitetura de microprocessadores. Este artigo também apresenta a questão de processadores projetados para ser eficientes energeticamente ou de baixo consumo.

## 2. POTÊNCIA

Recentemente, o alto desempenho de microprocessadores embarcados em plataformas com bateria requer um processador de baixíssimo consumo. Com o avanço tecnológico na área de semicondutores, hoje em dia, a maioria dos circuitos digitais é construída usando os circuitos CMOS. Por isso, para enfrentar os entraves quanto ao consumo de potência nos processadores, é preciso analisar as fontes de dissipação e entender o que causa a dissipação dentro do transistor. O consumo de potência no transistor CMOS é categorizado em dois tipos: consumo de potência dinâmica e consumo de potência estática.

$$P_{\text{total}} = P_{\text{dynamic}} + P_{\text{static}}$$

$$P_{\text{total}} = \alpha f (1/2) C V_{DD}^2 + t_{sc} V_{DD} I_{\text{peak}} f + \sum V_{DD} I_{\text{leakage}} f$$

Load Charge      Short Circuit      Leakage

Figura 1. Equação de potência total.

O consumo dinâmico é aquele associado às transições entre os níveis de tensão nas entradas e saída das portas. Quando estas transições acontecem, o consumo dinâmico ocorre e ele é composto pela soma do consumo de carga e consumo de curto-circuito. O consumo de potência dinâmico é calculado na equação abaixo:

$$P_{\text{dynamic}} = \alpha f (1/2) C V_{DD}^2 + t_{sc} V_{DD} I_{\text{peak}} f$$

Figura 2. Equação de potência dinâmica.

Consumo de carga é a energia necessária para carregar a capacitância nas portas de saída. Consumo de curto-circuito é a energia que flui entre VDD e GND quando há transições nas portas de entrada que resultam em um caminho entre a rede pull-up (transistores PMOS) e a rede pull-down (transistores NMOS) simultaneamente ligados.

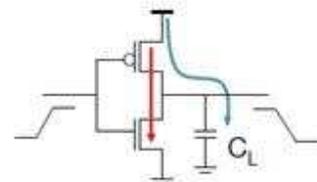


Figura 3. Modelo de consumo de um inversor durante chaveamento.

Consumo estático é aquele associado à energia que flui entre VDD e GND independentemente se há ou não transição alguma

nos sinais. O consumo de potência estático é calculado na equação abaixo:

$$P_{\text{static}} = \sum V_{DD} I_{\text{leakage}}$$

Figura 4. Equação de energia estática.

Este consumo estático existe devido às correntes de fuga que são correntes elétricas que fluem através de dois pontos em que deveria haver idealmente uma resistência infinita. Atualmente, o consumo estático se tornou um componente de consumo considerável, tornando-se um problema nas tecnologias avançadas, pois as correntes de fuga aumentaram nas menores dimensões (Ex: 90nm, 65nm, 45nm).

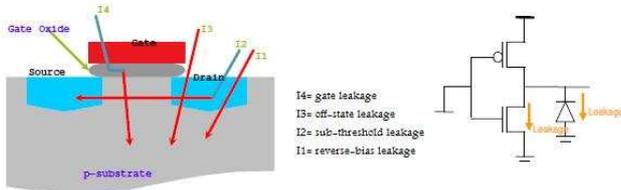


Figura 5. Fontes de fuga de energia em um transistor.

### 3. POTÊNCIA VERSUS ENERGIA

No contexto de processadores de baixo consumo, nem sempre o núcleo que desempenha uma determinada tarefa com menor consumo de energia é o mais adequado para uma dada aplicação. Suponha os gráficos abaixo como o perfil de consumo de dois processadores de diferentes arquiteturas realizando a mesma função.

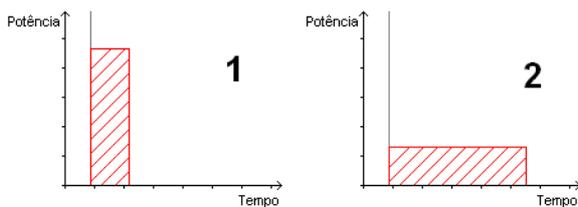


Figura 6. Perfil de consumo para processadores 1 e 2 executando a mesma tarefa.

Embora o consumo de energia (área do gráfico em vermelho) seja o mesmo, portanto igualmente energy-efficient, o processador 2 é notadamente low-power em relação ao processador 1. Porém nota-se que o processador 1 possui maior desempenho que o processador 2, já que conclui a tarefa em menor tempo.

A escolha do melhor processador dentre os dois apresentados está diretamente relacionada à aplicação que se dará aos mesmos. Embora o processador 1 seja ótimo em uma aplicação que demanda alto desempenho, ele não será adequado se o sistema em questão não suportar o surto de corrente que este perfil de consumo irá gerar ou o montante de calor que este processador irá

dissipar. Ao processador 2, convém as aplicações de baixíssimo consumo como operações com energy-harvesting ou os sistemas que operam sem auxílio de dissipadores de calor.

### 4. Potência versus Energia na Prática: Dois Cores em comparação

Qual o melhor processador de baixa potência para uma determinada aplicação? Nem sempre processadores de baixa potência são os mais eficientes energeticamente e muitas vezes isto é proposital e desejável.

O caso de um processador que opere com energia obtida do ambiente (energy harvesting) é um ótimo exemplo. Ele estará recebendo energia a todo o instante e, de uma forma geral, não será capaz de acumular tudo o que não consumir, portanto ainda que este processador execute tarefas que potencialmente sejam um desperdício de energia, como uma tomada de branch incorreto e posterior reversão, desde que não ultrapassando o envelope de energia fornecido, este processador continuará sendo low-power e continuará sendo adequado à tarefa a que se destina ainda que não seja energy efficient.

Um exemplo o qual este conceito se aplica está na comparação descrita no artigo “Low Power Case Study #1: Actual Algorithm Execution” [3] da MicroChip. Note que embora o MSP430 apresente desempenho inferior e consuma mais energia que o PIC24, ele executa toda a tarefa sempre com menor consumo instantâneo.

É evidente que o artigo em questão visa beneficiar o PIC24, já que foi criado pela MicroChip, o artigo “Ultra-Low Power Comparison: MSP430 vs. microchip XLP Tech Brief” [4] da Texas Instruments, por exemplo abordará um ponto de vista oposto e evidentemente benéfico ao MSP430.

Porém o que deve ser ressaltado e é nítido em ambos os contextos é que cada produto se destina a uma dada aplicação específica. Um deles pode não ter seu foco principal voltado à questão de eficiência energética, porém ser extremamente low-power, já o outro pode ter grande foco na eficiência energética e não ser low-power. Cabe, portanto ao designer do sistema adotar o core adequado ao que se pretende buscar em cada caso.

### 5. Métodos para economizar potência/energia

Embora seja sabido que decisões corretas de arquitetura são a melhor forma de se economizar energia em um determinado contexto, outras técnicas também podem ser empregadas em vários outros níveis de abstração durante o desenvolvimento. Algumas das mais comuns são:

#### 5.1 Redução da frequência de operação

Se o sistema permite um desempenho mais baixo, é recomendável diminuir a frequência de operação ao máximo para economizar potência dinâmica, vide equação da figura 1. A redução no consumo de potência dinâmica sempre é linearmente proporcional com a redução da frequência de operação, mas é razoável esperar que o efeito desta técnica varie de acordo com as particularidades de cada circuito. O consumo dinâmico de um circuito não necessariamente cai pela metade se a frequência for diminuída pela metade, especialmente se o circuito já tem técnicas aplicadas como clock gating. Além disto, é comum que em um sistema complexo não seja possível abaixar o clock geral do sistema, porque apenas algumas partes do circuito poderiam trabalhar a

freqüência mais devagar. Nestes casos, o uso de múltiplas freqüências em múltiplos domínios de clock também é um artifício possível para manter o consumo mais baixo possível.

## 5.2 Redução da tensão de operação

Este método reduz tanto o consumo de chaveamento (de forma quadrática) quanto o consumo estático (de forma direta), porém esta redução de nível de tensão possui um impacto direto no desempenho do circuito. Quando considerado um mesmo nó tecnológico quanto menor a tensão, mais lentas serão as portas lógicas que compõe o circuito.

## 5.3 Múltiplas tensões de operação

Atualmente, um circuito para aplicações de baixo consumo pode receber uma única tensão de entrada e conter várias tensões de operação internamente. Esta técnica permite diminuir o consumo de potência localmente, onde é possível trabalhar com menor tensão. Em um circuito complexo, é comum exigir que alguns trechos do circuito tenham mais desempenho que outros. Os segmentos do circuito que estão fora do caminho crítico podem operar com portas lógicas mais lentas e, portanto, em uma menor tensão consumindo menos. Esta técnica de baixa potência exige o uso de uma célula especial (level shifter) que não corrompe o dado ao cruzar o domínio de tensões, convertendo o nível de tensão do dado lógico. Esta técnica de múltipla tensão contribui para o baixo consumo dinâmico e estático.

## 5.4 Circuito com portas lógicas de alto Vt

Com o escalonamento de processo, o consumo estático tem virado o maior contribuinte no consumo total de potência. Nos processadores modernos, quase 50% do consumo são do componente estático (leakage). Portas lógicas de alto Vt é uma das únicas técnicas que focam no consumo estático. Esta técnica se baseia em usar portas lógicas especiais que são construídas para ter pouca corrente de fuga. Normalmente, estas portas lógicas especiais são providas pela biblioteca da fabricante do chip. Estas portas lógicas são compostas por transistores que têm o limiar mínimo de tensão mais alto (high-Vt), resultando em menos corrente de fugas e economizando o consumo de potência estático. Em contrapartida, estes transistores com alto Vt são mais lentos e só podem ser utilizados em caminhos não-críticos, resultando em um sistema com transistores de múltiplos Vt. Vide [5].

## 5.5 Escalonamento de tensão e freqüência dinamicamente

O uso desta técnica de baixo consumo é complicado, mas o seu conceito é simples. Da mesma maneira que o uso de múltiplas tensões e freqüências é aplicado para reduzir consumo de potência em troca de perda de desempenho, o escalonamento dinâmico de tensão e freqüência faz o mesmo. A diferença é que as técnicas anteriores são aplicadas espacialmente no circuito, enquanto esta técnica de escalonamento dinâmico é aplicada temporalmente no circuito. Isto quer dizer que o mesmo trecho de circuito pode operar em duas freqüências ou tensões diferentes, dependendo do momento em que se encontra. Para isto ocorrer, além das células level-shifter, um controlador de modos de operação reduz o consumo do chip por demanda ao abaixar a tensão e freqüência de cada trecho quando a aplicação não exige um desempenho alto.

Aplicativos móveis como celular é um perfeito exemplo de uso desta técnica. Vide [6].

## 5.6 Codificação dos sinais de barramento

Em qualquer processador, seja arquitetura Von Neuman ou Harvard, o barramento entre o núcleo e memória normalmente tem alta capacitância e consome a maior parte do sistema. Esta técnica consiste em minimizar o número de transições dos sinais de barramento. Existem diferentes métodos de codificação, cada um com seus prós e contras de acordo com a complexidade do codificador, com o tipo de barramento, o tipo de acesso, e arquitetura do barramento. Um famoso exemplo desta técnica é o método "bus-invert" que utiliza a distância de Hamming do dado atual com o próximo para inverter os sinais. Outro exemplo é a codificação Gray que faz proveito do princípio da localidade espacial que ocorre ao processador ao acessar o barramento de endereço. Vide [7].

## 5.7 Paralelismo na medida certa:

### 5.7.1 Ajustando o pipeline depth

O número de estágios de um pipeline (pipeline depth) de um processador depende do ISA e da aplicação. O nível de paralelismo por instrução (ILP) muitas vezes não é aproveitado pelo software, resultando em um overhead de hardware à toa que não traz grandes melhoras de desempenho. Este overhead de hardware consiste em registradores que foram adicionados para criar os novos estágios de pipeline. Para evitar o consumo destes registradores extras do pipeline, existe um ponto ótimo entre desempenho e consumo que precisa ser avaliado para cada combinação de aplicação e ISA. Para aplicações de baixo desempenho e consumo, um ótimo exemplo é o processador da ARM Cortex M0. Este processador não é super escalar, nem SIMD e nem tem predição de branch. Ele é apenas um RISC de 32-bits, arquitetura Von Neuman e com pipeline de apenas 3-estágios. Vide [8].

### 5.7.2 Eliminação de estágios de pipeline

Uma técnica conhecida como pipeline stage unification (PSU) [7] unifica múltiplos estágios de pipeline de forma dinâmica, encurtando o pipeline depth quando apropriado. Para que isto ocorra alguns registradores de pipeline são evitados. Esta técnica de PSU economiza energia de duas formas. Ao parar o sinal do clock para os registradores de contorno, reduz a capacitância de carga no clock, diminuindo o consumo. Ao reduzir o número de ciclos para executar todos os estágios do pipeline, diminui o consumo também. Outra técnica, esta bem antiga, é o stage-skip pipeline [2] que utiliza um buffer dentro do processador para armazenar sinais de controle de um loop já decodificado. Ao executar, por exemplo, um loop que contém repetidamente a operação multiply-accumulate como em rotinas multimídia, o stage-skip pipeline reduz o consumo de potência ao eliminar os estágios de fetch e decode. Sem precisar acessar a memória de instruções, o stage-skip pipeline apenas executa a operação continuamente como especificado pelos sinais do buffer.

### 5.7.3 Não utilização de predição de branch

Para garantir as vantagens do paralelismo em nível de instrução, é necessário ter técnicas especulativas para prever quando um branch acontece. O problema da predição de branch para processadores energy efficient reside no fato que toda a energia gasta pela especulação pode ser desperdiçada. Em processadores

cujo objetivo principal é a economia de energia, a predição de branch não é recomendável; uma vez que quanto maior a taxa de erro, maior será o consumo não utilizável do core. Além disso, processadores de baixo consumo são simples, com pipeline depth pequeno (até 3 estágios apenas), por isso a técnica de predição de branch se torna menos atrativa; uma vez que quando a predição é feita corretamente, economiza-se poucos estágios de pipeline. No entanto, para contornar dinamicamente este desperdício de energia, pode-se usar uma técnica chamada pipeline gating [10]. Esta técnica adiciona um overhead de circuito para analisar a confiança e qualidade da estimativa feita. Este circuito adicionado decide se deve fazer o “gating” dos próximos estágios de pipeline que são relacionados à predição de branch. Ao parar o pipeline, criam-se os “stalls”, evitando os estágios especulativos e economizando a energia que seria provavelmente desperdiçada.

## 5.8 Design assíncrono

A indústria adotou a lógica síncrona como padrão ao desenvolvimento de circuitos, mas a técnica de design assíncrono, quando empregada corretamente, pode levar a enormes ganhos de desempenho e economia de energia. É comum um chip até ter 40% de consumo devido à distribuição de clock pelo seu circuito. A lógica assíncrona dispensa toda esta árvore de clock que consome muito. Além do mais, o design assíncrono tem maior tolerância às condições de operações e variações de processo e não precisa se preocupar com balanceamento de clock. Porém, vale lembrar nem todos os tipos de problemas podem se beneficiar desta técnica menos convencional e que existem menos ferramentas de suporte para lógica assíncrona.

## 5.9 Clock Gating

Uma das mais comuns hoje em dia, o clock gating consiste basicamente em “parar” o clock em determinados momentos e porções do hardware. Esta parada leva a extinção do consumo dinâmico nos componentes que estão sob sua influência. Atualmente, as ferramentas de síntese lógica conseguem inferir no código RTL quais flip-flops podem ter o seu clock temporariamente parado e automatizar esta técnica de baixo consumo.

## 5.10 Power Gating ou Power Shut-off

Mais agressiva e menos utilizada do que o clock gating a utilização de domínios de tensão que podem ser desligados reduz além do consumo dinâmico, o consumo estático, levando o circuito sob sua influência a um estado nulo de consumo de energia. Esta técnica exige um controlador de modos de operação que nunca é desligado, mas que pode desligar os outros blocos quando a aplicação permitir. Esta técnica também exige células especiais de isolamento e de retenção. Como a maioria dos blocos é interconectada, a célula especial de isolamento impede que o bloco desligado não corrompa o bloco ligado ao transmitir um sinal imprevisível. A célula especial de retenção salva o estado do bloco quando este é desligado e garante que ao religar o bloco, ele retorne como antes.

## 5.11 Microprocessadores com vários modos de operação

O microprocessador de baixo consumo precisa ser orientado às interrupções e ter vários modos de operação, os quais permitem desligar os periféricos ou o próprio núcleo que não estão sendo utilizados. É sabido que os programas voltados a baixo consumo são totalmente orientados à interrupção, pois estas aplicações

focam em colocar o microprocessador ao máximo de tempo no modo sleep e deep-sleep, deixando à ativa apenas os componentes em processamento. Um exemplo de microprocessador com vários modos low-power é a família MSP430. Este microprocessador tem cinco modos de operação, sendo que o modo ativo tem corrente de 220  $\mu$ A e modo mais low-power tem uma corrente de 0.1  $\mu$ A. Vide [9].

## 5.12 Wake-up rápido

O microprocessador de baixo consumo precisa ter um wake-up rápido. Se o sistema tiver uma longa latência para acordar, ele vai ficar menos tempo no modo low-power. Em outras palavras, quanto mais rápido um MCU acorda, mais tempo terá para dormir e economizar. O tempo de wake-up é a latência necessária para estabilizar o sistema e prepará-lo antes de ele entrar à ativa. Por exemplo, esta latência de wake-up inclui estabilização do PLL ou oscilador de clock. Se o sistema não acordar a tempo, ele pode perder algum evento em tempo real, portanto é muito importante garantir este tempo de wake-up.

## 6. Conclusão

Alcançar alto desempenho e baixo consumo em microprocessadores modernos é um constante desafio no desenvolvimento do processador. Muitas técnicas de arquitetura do processador exigem o entendimento sistêmico e visão geral do produto, pois estas decisões de trade-off entre o desempenho e energia requerem muita interação com o software e aplicação.

Este artigo resumiu várias técnicas para se atingir baixa potência e eficiência energética em processadores além de abordar e exemplificar diferentes conceitos e aplicações para cores com características low-power e/ou energy efficient.

Em suma, este artigo abordou o escalonamento dinâmico e estático de frequência e tensão, uso de clock gating, power gating, design assíncrono, paralelismo na arquitetura de pipeline do processador dentre outras técnicas extremamente pertinentes ao desejo de se projetar circuitos extremamente econômicos.

## 7. Agradecimentos

Gostaríamos de agradecer aos autores dos livros e artigos que nos referenciamos para criar este material, além de nosso professor e universidade que nos deram oportunidade para nos aprofundarmos no mundo dos processadores Low-Power e Energy Efficient dos quais temos muito gosto.

## 8. Referências

- [1] M. Keating, D. Flynn, R. Aitken, A. Gibbons, K. Shi; Low Power Methodology Manual For System-on-Chip Design.
- [2] M. Hiraki, R. S. Bajwa, H. Kojima, D. J. Gorny, K. Nitta, A. Shridhar, K. Sasaki, K. Seki; Stage-Skip Pipeline: A low Power Processor Architecture Using a Decode Instruction Buffer.
- [3] Low Power Case Study #1: Actual Algorithm Execution; [www.microchip.com/XLP](http://www.microchip.com/XLP)
- [4] Ultra-Low Power Comparison: MSP430 vs. microchip XLP Tech Brief; <http://focus.ti.com/lit/wp/slay015/slay015.pdf>
- [5] Diary R. Sulaiman; Microprocessor Leakage power reduction using dual supply voltage scaling; 5th International Advanced Technologies Symposium (IATS'09), 2009

- [6] Kazuaki Murakami and Hidetaka Magoshi; Trends in High-Performance, Low-Power Processor Architectures; IEICE Trans. Electron., Vol.E84-C, No.2 February 2001
- [7] Muhammad Yasir Qadri, Hemal S. Gujarathi, Klaus D. McDonald-Maier; Low Power Processor Architectures and Contemporary Techniques for Power Optimization – A Review; Journal of Computers, Vol. 4, No. 10, October 2009
- [8] Cortex-M0 Devices - Generic User Guide; ARM; [http://infocenter.arm.com/help/topic/com.arm.doc.dui0497a/DUI0497A\\_cortex\\_m0\\_r0p0\\_generic\\_ug.pdf](http://infocenter.arm.com/help/topic/com.arm.doc.dui0497a/DUI0497A_cortex_m0_r0p0_generic_ug.pdf)
- [9] Mixed Signal Microcontroller MSP430F20xx; <http://focus.ti.com/lit/ds/slas491g/slas491g.pdf>
- [10] S. Manne, A. Klauser, and D. Grunwald; Pipeline gating: speculation control for energy reduction; In Proceedings of the 25th Annual International Symposium on Computer Architecture 1998