

O conteúdo do presente relatório é de única responsabilidade do(s) autor(es).
(The contents of this report are the sole responsibility of the author(s).)

**Minimização do Consumo de Energia em
um Sistema para Aquisição de Dados
Controlado por Microprocessador¹**

*Paulo Cesar Centoducatte
Nelson Castro Machado*

Relatório Técnico DCC-10/93

Abril de 1993

Minimização do Consumo de Energia em um Sistema para Aquisição de Dados Controlado por Microprocessador[†]

Paulo Cesar Centoducatte[‡]
Nelson Castro Machado[§]

Sumário

O presente trabalho apresenta algumas técnicas específicas, que podem ser aplicadas, em conjunto com as tradicionais, para a redução de consumo de energia em equipamentos baseados em microprocessadores, que tenham como característica de operação ciclos de atividades separados por ciclos de espera sem que algum trabalho útil seja realizado. Estas técnicas foram aplicadas no desenvolvimento de um sistema de aquisição de dados meteorológicos, possibilitando o uso de baterias em experimentos de longa duração em locais remotos não providos de energia elétrica.

1 Introdução

Um requisito frequente no projeto de equipamentos controlados por microprocessadores é a alimentação por baterias, seja porque se tratam de equipamentos portáteis, seja porque devem operar em locais não servidos por energia elétrica. Este tipo de projeto exige, por um lado a minimização do consumo para aumentar a autonomia de operação e por

[†]Projeto parcialmente financiado pelo CNPq e CAPES.

[‡]DUCATTE@dcc.unicamp.br

[§]MACHADO@dcc.unicamp.br

outro a introdução de técnicas de “backup” de memória que garantam a não volatilidade, mesmo quando do esgotamento da bateria principal, de pelo menos uma parte crítica da memória. Duas estratégias óbvias são inicialmente adotadas para diminuir o consumo: redução, ao máximo, do número de componentes e utilização quando, disponíveis, de componentes de baixo consumo, especialmente os fabricados em tecnologia CMOS. Os dois objetivos acima estabelecidos exigem também a adoção de algumas técnicas especiais de projeto, especialmente no caso de equipamentos baseados em microprocessadores caracterizados por operação intermitente. Entende-se como operação intermitente aquela em que dois períodos de atividades são separados por um período onde nenhum trabalho útil é realizado. Exemplo de equipamentos com este comportamento são estações de aquisição de dados meteorológicos, que após um ciclo de coleta de dados podem permanecer um longo período de tempo à espera de um novo ciclo.

Este trabalho descreve algumas destas técnicas especiais de projeto, adotadas na implementação do SADAM — Sistema de Aquisição de DADOS Meteorológicos, projetado para possibilitar a realização de experimentos de longa duração (um mês) em locais remotos, onde não há energia elétrica disponível. Acreditamos ser esta discussão de interesse de projetistas de hardware e arquitetos de sistemas, de vez que é bastante difícil encontrar na literatura um tratamento detalhado de tais recursos. Segue-se uma breve descrição da estação de campo SADAM e a discussão das técnicas adotadas para minimizar o consumo de energia e garantir a não volatilidade de parte da memória. Uma descrição completa das características e implementação do sistema SADAM, Hardware e Software pode ser encontrada em Centoducatte[1] e Gimenes[3], respectivamente.

2 A Estação de Campo SADAM

A figura 1 mostra em forma de diagrama de blocos a Estação de Campo do SADAM cujo hardware inclui recursos mínimos para permitir programação e intervenção do operador durante a realização de um experimento. O “coração” da estação é a placa principal, que contém a

maioria dos principais elementos do sistema, incluindo o microprocessador 80C188, 8 KB (kilobytes) de memória de acesso aleatório (RAM), 16 KB de memória de leitura apenas (EPROM), relógio de tempo real MM58167A, circuito de controle da alimentação, adaptador para teclado (destinado à entrada de dados ou comandos), adaptador para “display” de cristal líquido — LCD (que permite comunicação com o experimentador) e “slots” destinados à conexão de um cartucho removível de RAM, conversor analógico/digital, sensores e expansões¹.

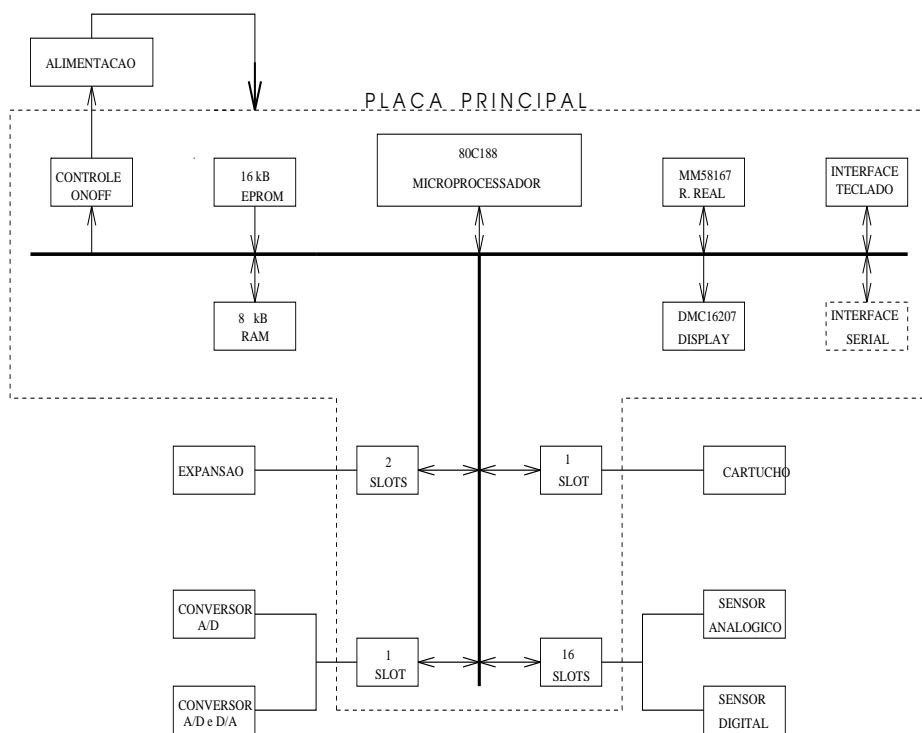


Figura 1: Diagrama de blocos da Estação de Campo

¹Foi ainda adicionada, ao protótipo da Estação de Campo, uma interface serial como recurso de depuração

Além das técnicas habituais de redução do número de componentes e utilização, quando disponíveis, de versões CMOS dos integrados, os principais recursos para minimização do consumo, utilizados no projeto da estação de campo, foram:

- a adoção de dois modos de operação, denominados de modo normal ou ativo e “standby” ou inativo;
- utilização como meio de armazenamento de dados um cartucho de RAM-CMOS com “backup” de bateria em substituição a outros meios de armazenamento de maior consumo, tais como fita casete, disquetes, etc. . . e dotando a estação de maior flexibilidade quanto ao local de instalação, quando comparado com a utilização de linha serial para transmissão dos dados coletados para a estação base.

A primeira técnica consiste no seguinte: quando a estação está ativa ou no modo normal todos os seus componentes estão energizados; este modo de operação é utilizado durante a realização dos ciclos de aquisição de dados ou durante uma interação com o experimentador por ele solicitada via teclado. No modo “standby”, apenas são energizados uns poucos componentes essenciais — os demais são “desligados”; a estação entra automaticamente neste modo de operação após concluído um ciclo de aquisição ou a intervenção do operador. A estação retorna também automaticamente ao modo ativo, toda vez que um novo ciclo de aquisição deve ser executado ou uma intervenção é solicitada pelo operador. No modo “standby” somente os componentes estritamente necessários ao retorno da estação ao modo normal e aqueles destinados a preservar o contexto do experimento são mantidos alimentados — até o microprocessador é desenergizado.

A implementação desta estratégia está baseada na fonte de alimentação e no bloco “controle OnOff” da figura 1. Na figura 2 é apresentado um diagrama da fonte de alimentação que é composta por uma bateria, reguladores de tensão e chaves controladas pelo bloco “OnOff”, dotando a fonte de dois tipos de tensão de alimentação: tensão “On”, onde a alimentação está sempre presente, utilizada para alimentar os componentes

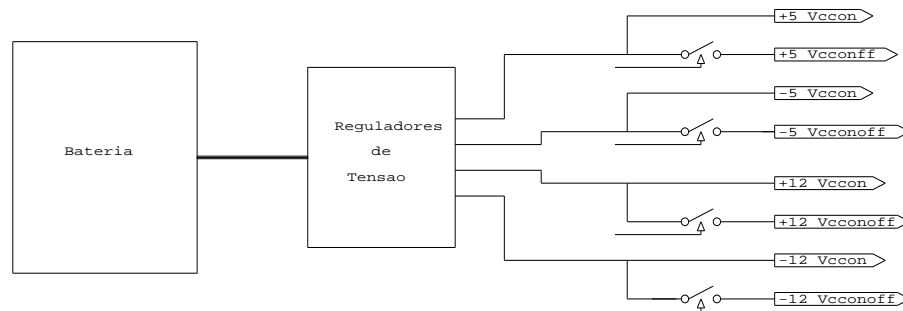


Figura 2: Fonte de alimentação

que devem permanecer sempre energizados e tensão “OnOff” que pode ser ligada ou desligada, utilizada para alimentar os componentes que são desenergizados quando a estação entra no modo de operação “standby”.

Outra característica essencial da estação de campo do SADAM é a utilização de “cartuchos” de RAM, que são módulos removíveis contendo uma certa quantidade de memória RAM (tipicamente, de 64 KB a 1 MBytes) alimentados por bateria local durante o transporte e pela bateria principal (VccOn) da estação durante a realização do experimento de coleta de dados. O aspecto de maior interesse no projeto dos cartuchos é a utilização de circuitos que garantam o chaveamento seguro entre as duas fontes de alimentação, garantindo a integridade da RAM e protegendo-a contra escritas espúrias.

3 Sistema de Alimentação

Os requisitos de alimentação do SADAM são tensões de +5 Vcc, +12 Vcc e -12 Vcc. As tensões de 12 Vcc são necessárias para o sistema de Conversão Analógico/Digital e para as interfaces SADAM/sensores, pois alguns sensores que podem vir a serem utilizados fazem uso destes níveis de tensão.

O principal recurso usado foi manter somente o mínimo necessário

de dispositivos alimentados, quando a estação entra em um período “longo”² sem aquisição de dados. Neste período a estação está em “standby”. Ao se manter o sistema em “standby”, consegue-se minimizar bastante o consumo do SADAM, principalmente quando da realização do experimento padrão em meteorologia, caracterizado por um período de coleta de dados em torno de 30 dias. Contudo, os ciclos de aquisição de dados são separados por um “longo” período sem aquisição. Isto possibilita que baterias de médio ou pequeno porte, similares às encontradas em automóveis ou motos, sejam empregadas na alimentação do SADAM. Obviamente são utilizadas versões seladas de tais baterias, apropriadas para uso em instrumentação.

Para a implementação do sistema de “standby” é necessário que o sistema de alimentação forneça uma tensão sempre presente, denominada de “V_{con}” e outra controlada, por um processo digital, que hora está disponível e hora não, denominada “V_{conoff}”. Assim quando o SADAM entra em “standby”, alguns de seus componentes não deixam de ser alimentados, como é o caso da memória RAM, do relógio e de alguns componentes das interfaces SADAM/sensores que guardam valores intermediários. Por outro lado, existem diversos componentes que podem ser desenergizado, sem afetar o resultado final do experimento como é o caso da CPU, da memória EPROM e da maioria dos periféricos do SADAM e das interfaces.

4 Implementação do Sistema de Alimentação OnOff

Todo o controle do sistema de alimentação “OnOff” está baseado no chip de relógio de tempo real MM58167A, que possui dois sinais de interrupção programáveis por Software. Quando o intervalo de tempo para o próximo ciclo de aquisição for “longo”, o relógio é programado para

²Longo aqui é caracterizado como sendo um período de tempo maior que o tempo necessário para todos os componentes do sistema estarem prontos a realizarem suas funções após o sistema ser ligado. No SADAM 1 minuto é considerado “longo”.

gerar um pedido de interrupção ao chegar a hora de um novo período de aquisição. O “status” da CPU é guardado na RAM e a alimentação “OnOff” é desligada. Quando chega a hora do novo período de aquisição, o MM58167A gera um pedido de interrupção e este sinal é utilizado por um circuito auxiliar permanentemente alimentado, a fim de ligar a alimentação “OnOff”, tornando assim todo o sistema apto a executar mais um ciclo de aquisição de dados.

Para desempenhar as funções acima mencionadas, o relógio MM58167A possui um estado “PowerDown”, no qual consome muito menos energia que no estado normal de funcionamento e não responde a nenhum comando externo enquanto perdurar o “power-down”. Isto é muito importante, pois a utilização desta característica permite evitar que comandos espúrios sejam executados pelo relógio durante o processo de estabilização do sistema, após a alimentação “On-Off” ser ligada. Um comando espúrio no relógio poderia causar a perda da hora corrente, o que arruinaria o experimento.

A figura 3 mostra o circuito de controle da alimentação “OnOff”. Dois “flip-flops” tipo D são utilizados para controle da alimentação: o “flip-flop” A é controlado pela interrupção do relógio de tempo real e o “flip-flop” B permite que a alimentação seja colocada no estado “On” através do teclado. Ambos os “flip-flops” são mapeados no espaço de E/S, permitindo assim controle pelo programa. Segue-se uma descrição detalhada da operação do circuito.

Um dos problemas a ser evitado é a execução de um comando espúrio pelo MM58167A: durante a permanência da alimentação “OnOff” em “Off”, o MM58167A está energizado e deve ser mantido no estado “PowerDown” por ação de um sinal externo. Se este sinal externo, quando a alimentação passa de “Off” para “On”, retirar o MM58167A do estado de “PowerDown” antes que os sinais do barramento estejam estáveis, o relógio poderia executar um comando espúrio; no pior caso, este comando espúrio poderia alterar os valores dos registradores internos do relógio. A probabilidade de tal situação ocorrer é muito pequena, porém existe. Para solucionar este problema é necessário garantir que \overline{WR} e \overline{CS} do MM58167A já estejam estáveis quando o relógio sair de “PowerDown”.

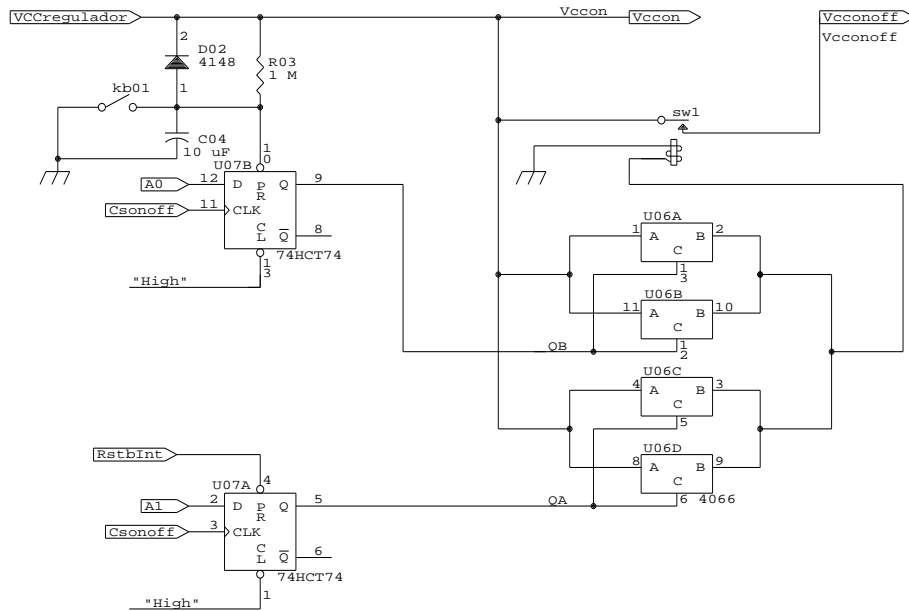


Figura 3: Controle da alimentação “On-Off”

Para controlar o pino $\overline{PowerDown}$ do MM58167A foi utilizado um “flip-flop” do tipo D mapeado no espaço de E/S e que pode ser ligado ou desligado pelo programa. Durante o período de estabilização do sistema, o sinal de “Reset” aplicado a este “flip-flop” garante a permanência em “baixo” do sinal que chega ao pino $\overline{PowerDown}$, não permitindo que o MM58167A saia do estado de “PowerDown”. O MM58167A só é retirado do estado de “PowerDown” para ser reprogramado ou consultado e em seguida volta a este estado. A figura 4 mostra o esquema utilizado.

Um outro problema está associado ao fato da alimentação da CPU ser desligada durante o estado “standby”: consiste em se distinguir um reset “verdadeiro” de um reset “falso”. O reset “verdadeiro” é aquele que ocorre quando o sistema é ligado pela primeira vez e o “falso” é

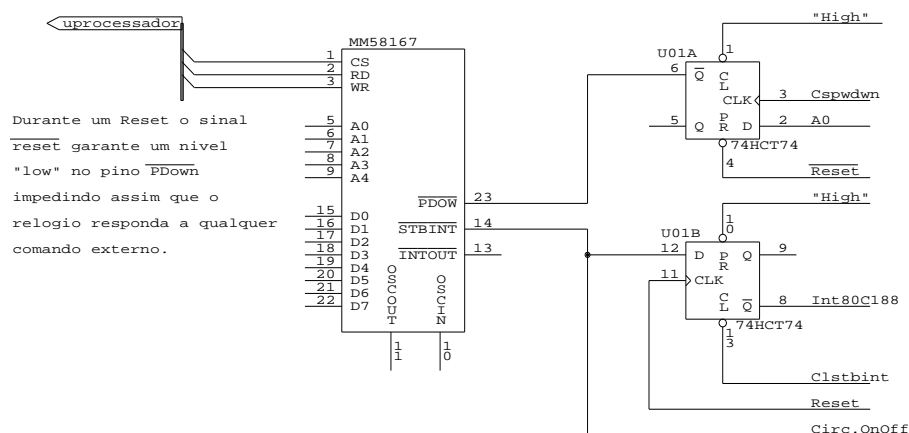


Figura 4: Controle de PowerDown do MM58167A e do pedido de interrupção standby

devido à volta de um estado “standby”. No reset “verdadeiro” devem ser executados todos os procedimentos de iniciação, tais como: verificar se a instalação dos sensores corresponde à programada; acertar a hora corrente; zerar os contadores dos sensores digitais; etc. e iniciar a aquisição dos dados propriamente dita. No reset “falso” somente deve ser restaurado o contexto da CPU e dado prosseguimento à aquisição dos dados.

A identificação de um reset pode ser feita tanto por software como por hardware. A solução por software pode ser implementada da seguinte forma: durante a programação do experimento, na Estação Base é gravado um padrão de bits no cartucho e em uma posição de memória conhecida. Quando um reset ocorre, esta posição é verificada e se nela estiver o padrão colocado na Estação Base, então o reset é “verdadeiro” e os procedimentos de iniciação adequados serão executados e este padrão será substituído por um novo indicando para novas ocorrências um re-

set “falso”. Caso contrário será um reset “falso” e os procedimentos relacionados com este reset serão executados.

A solução por hardware pode ser implementada adicionando-se um “flip-flop” (U01B na figura 4) que faz com que, ao ocorrer um “reset verdadeiro”, o pedido de interrupção “standby” do MM58167A esteja desativado e ao surgir um “reset falso”, esteja ativo. Assim, se este pedido de interrupção “standby” for usado para gerar um pedido de interrupção à CPU, e se entre as primeiras instruções executadas, estiver a habilitação do sistema de interrupção da CPU, ao ocorrer um “reset verdadeiro” não haverá pedido de interrupção e a rotina de iniciação será executada. Se o “reset” for “falso”, haverá um pedido de interrupção, que será atendido por uma rotina que troca, na pilha, o endereço de retorno pelo endereço da rotina que trata o “reset falso” e retorna.

É importante salientar que a solução por hardware não implica na utilização de componentes adicionais e tem a vantagem de permitir o uso de um botão de reset de forma similar aos existentes nos micros pessoais de propósito geral, o que não é possível de forma tão simples na solução por software. Desta forma, foi adotada no SADAM a solução por hardware. Esta solução também é mostrada na figura 4.

5 Implementação do Cartucho

O cartucho de RAM com “backup” de bateria foi implementado de forma a garantir a integridade dos dados, isto é, ausência de perda ou adulteração parcial ou total dos dados na introdução/remoção do cartucho em uma das estações ou durante o transporte do mesmo. Estas perdas podem ocorrer devido à falta de alimentação da memória durante o processo de introdução/remoção ou ainda pela presença de um ou mais comandos de escrita espúrios gerados durante o mesmo processo.

Para se garantir alimentação contínua às RAM's que compõem o cartucho, durante seu transporte entre estações, incluiu-se a bordo uma pequena bateria de lítio, conectada como mostrado na figura 5, onde:

Onde:

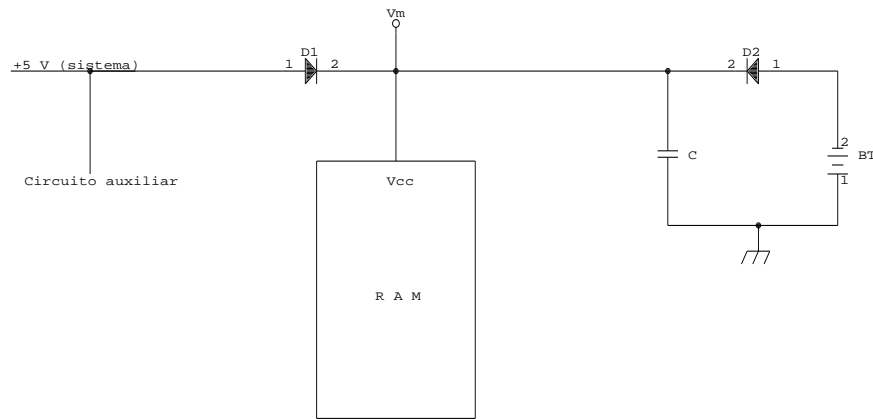


Figura 5: Alimentação contínua de memórias RAM

- V_{cc} é a tensão proveniente da estação (+5V)
- B é a bateria de lítio, com tensão menor que V_{cc} e maior que a mínima necessária para a manutenção dos dados pela RAM (tipicamente 3V)
- D_1 evita que o circuito auxiliar seja alimentado quando V_{cc} não estiver presente, economizando energia da bateria
- D_2 isola a bateria do sistema quando V_{cc} estiver presente no caso da bateria não ser recarregável, caso contrário D_2 deve ser substituído por um resistor de valor apropriado

Quando o cartucho é retirado de uma das estações V_{cc} deixa de existir e o capacitor C descarrega-se até o valor de $V_B - V_{D_2}$. Este valor é mantido como alimentação da memória até que o cartucho seja novamente inserido em uma das estações. O capacitor C carrega-se até o valor de $V_{cc} - V_{D_1}$ e esta passa a ser a tensão de alimentação da RAM — supondo-se que D_1 é um diodo com baixa queda de tensão direta

(V_{D_1}), esta tensão deve se situar em um valor ainda admissível para o funcionamento normal dos “chips” envolvidos.

É importante salientar que a alimentação do cartucho é feita pela bateria quando ele não está conectado a uma das estações. Assim é desejável que as memórias consumam o mínimo possível para possibilitar um maior tempo de “sobre-vida” dos dados. Por isto no cartucho são utilizadas RAM’s CMOS especiais (LP) de baixíssimo consumo.

Evitar a ocorrência de comandos de escritas espúrios durante a introdução ou remoção do cartucho nas estações é mais complexo que resolver o problema da continuidade da alimentação. Durante estas operações, flutuações dos sinais nos barramentos de dados, dos endereços e de controle, poderiam gerar um comando de escrita que adultere o conteúdo de um ou mais bytes do cartucho. Assim, a memória deve estar desabilitada para a escrita ou qualquer outro comando durante a ocorrência destas flutuações nos sinais dos barramentos.

A proposta apresentada por Woodhull[13] tem como desvantagem a necessidade de intervenção manual a fim de garantir que a memória somente esteja habilitada a receber comandos após a inserção do cartucho, quando o sistema estiver estável.

A solução adotada neste trabalho utiliza a idéia contida na solução apresentada por Woodhull, isto é, atraso na capacidade da memória em responder comandos até que o sistema esteja estável. Porém, eliminando os “buffer’s” nos barramentos de endereços e dados, proporcionando maior economia da energia consumida pelo cartucho e implementando de forma automática o sistema de atraso da habilitação da memória. A figura 6 mostra de forma esquemática o circuito de controle automático da habilitação da memória.

Quando o cartucho está fora de uma das estações, somente a memória está energizada por $V_m = V_b - V_{D_1}$ e está desabilitada por efeito dos resistores de “pull-up” dos NAND’s “Open-Collector” ligados a V_m . Na inserção, V_m passa a ser aproximadamente V_{cc} antes que o circuito permita que a memória seja habilitada, pois T_1 é chaveado ON antes de T_2 . T_3 funciona como uma chave e D_3 acende quando a tensão V_{cc} está presente. Na remoção do cartucho os resistores de “pull-up” desabilitam

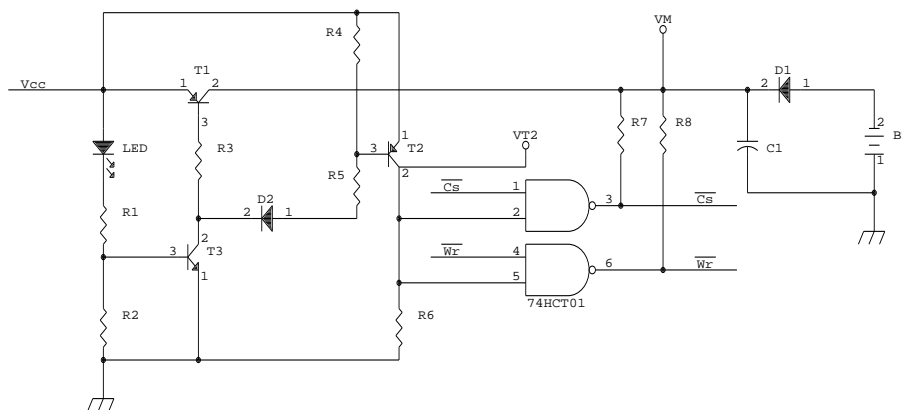


Figura 6: Circuito de controle automático da habilitação da memória

a memória assim que V_{cc} é removido.

6 Conclusões

A economia de energia proporcionada pela introdução dos dois modos de operação (ativo e inativo) é dependente do tempo em que a estação permanece no estado inativo, mas mesmo que este tempo seja somente uma fração do tempo em que ela permanece ativa, ainda assim pode valer a pena utilizá-los, pois quando a estação está no estado inativo somente cerca de 10% dos seus componentes estão alimentados. No SADAM, em experimentos que envolvem aquisições de dados meteorológicos algumas vezes ao dia, esta estratégia permite que o equipamento opere meses a fio, sem manutenção alguma em locais sem acesso a energia elétrica.

O uso do cartucho, como meio de armazenamento e transporte dos dados coletados não só propiciou maior economia como também mostrou-se bastante confiável, uma vez que em todos os testes de volatilidade e confiabilidade[1] a que foi submetido não houve perda ou adulteração de um único byte.

As técnicas aqui apresentadas podem ser utilizadas em qualquer sistema baseado em microprocessador com a inclusão de um relógio de tempo real capaz de gerar um pedido de interrupção de forma programada e de mais uns poucos componentes CMOS (“flip-flops” e portas básicas).

Referências

- [1] Centoducatte, P. C.. *SADAM — Um Sistema para Aquisição de Dados Meteorológicos*. Tese de Mestrado – DCC/IMECC – 1992.
- [2] Eggebrecht, L. C. . *Interfacing to the IBM Personal Computer*. Howard W. Sams & Co – First Edition, 1983 – Sixth Printing, 1988.
- [3] Gimenes, I. M. de S.. *Um Sistema Programável de Aquisição de Dados Meteorológicos*. Tese de Mestrado – DCC/IMECC – 1988.
- [4] Intel Corporation. *16-Bit Embedded Controllers Handbook*. Literature Department, 1990.
- [5] Intel Corporation. *iAPX 86/88, 186/188 User’s Manual Hardware Reference*. Literature Department, 1985.
- [6] Intel Corporation. *iAPX 86/88, 186/188 User’s Manual Programmer’s Reference*. Literature Department, 1985.
- [7] Intel Corporation. *Memory Components Handbook*. Literature Department, 1986.
- [8] Intel Corporation. *Microsystem Components Handbook – Microprocessors Volume I and II*. Literature Department, 1986.
- [9] Munich, S.. *The MM58167A Real Time Clock in a Battery Backed-up Design provides Reliable Clock and Calendar Functions*. National Semiconductor Application Note 359 – Feb. 1984.

- [10] National Semiconductor. *MM58167 Microprocessor Compatible Real Time Clock*. Eletronic Data Processing – 1984.
- [11] Schwartz, M.. *MM58167A Real Time Clock Design Guide*. National Semiconductor Application Note 353 – May. 1984.
- [12] Thomson, P. K.. *The MM58274 Adds Reliable Real Time Keeping to Any Microprocessor System*. National Semiconductor Application Note 365 – Apr. 1984.
- [13] Woodhull, A. S. *An EPROM Simulator*. Byte, March 1984, pp. 400–406.

Relatórios Técnicos – 1992

- 01/92 **Applications of Finite Automata Representing Large Vocabularies**, *C. L. Lucchesi, T. Kowaltowski*
- 02/92 **Point Set Pattern Matching in d -Dimensions**, *P. J. de Rezende, D. T. Lee*
- 03/92 **On the Irrelevance of Edge Orientations on the Acyclic Directed Two Disjoint Paths Problem**, *C. L. Lucchesi, M. C. M. T. Giglio*
- 04/92 **A Note on Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams**, *W. Jacometti*
- 05/92 **An (l, u) -Transversal Theorem for Bipartite Graphs**, *C. L. Lucchesi, D. H. Younger*
- 06/92 **Implementing Integrity Control in Active Databases**, *C. B. Medeiros, M. J. Andrade*
- 07/92 **New Experimental Results For Bipartite Matching**, *J. C. Setubal*
- 08/92 **Maintaining Integrity Constraints across Versions in a Database**, *C. B. Medeiros, G. Jomier, W. Cellary*
- 09/92 **On Clique-Complete Graphs**, *C. L. Lucchesi, C. P. Mello, J. L. Szwarcfiter*
- 10/92 **Examples of Informal but Rigorous Correctness Proofs for Tree Traversing Algorithms**, *T. Kowaltowski*
- 11/92 **Debugging Aids for Statechart-Based Systems**, *V. G. S. Elias, H. Liesenberg*
- 12/92 **Browsing and Querying in Object-Oriented Databases**, *J. L. de Oliveira, R. de O. Anido*

Relatórios Técnicos – 1993

- 01/93 **Transforming Statecharts into Reactive Systems**, *Antonio G. Figueiredo Filho, Hans K. E. Liesenberg*
- 02/93 **The Hierarchical Ring Protocol: An Efficient Scheme for Reading Replicated Data**, *Nabor das C. Mendonça, Ricardo de O. Anido*
- 03/93 **Matching Algorithms for Bipartite Graphs**, *Herbert A. Baier Saip, Cláudio L. Lucchesi*
- 04/93 **A lexBFS Algorithm for Proper Interval Graph Recognition**, *Celina M. H. de Figueiredo, João Meidanis, Célia P. de Mello*
- 05/93 **Sistema Gerenciador de Processamento Cooperativo**, *Ivonne. M. Carrazana, Nelson. C. Machado, Célio. C. Guimarães*
- 06/93 **Implementação de um Banco de Dados Relacional Dotado de uma Interface Cooperativa**, *Nascif A. Abousalh Neto, Ariadne M. B. R. Carvalho*
- 07/93 **Estadogramas no Desenvolvimento de Interfaces**, *Fábio N. de Lucena, Hans K. E. Liesenberg*
- 08/93 **Introspection and Projection in Reasoning about Other Agents**, *Jacques Wainer*
- 09/93 **Codificação de Seqüências de Imagens com Quantização Vetorial**, *Carlos Antonio Reinaldo Costa, Paulo Lício de Geus*

Departamento de Ciência da Computação — IMECC
Caixa Postal 6065
Universidade Estadual de Campinas
13081-970 – Campinas – SP
BRASIL
`reltec@dcc.unicamp.br`