

Aquisição e Análise de Sinais Elétricos

PEDRO RIBEIRO DE ANDRADE NETO
JOÃO CARLOS GIACOMIN (ORIENTADOR)

UFLA – Universidade Federal de Lavras
DCC – Departamento de Ciência da Computação
Cx. Postal 37, CEP 37.200-000 – Lavras (MG)
{pedron, giacomini}@comp.ufla.br

Resumo. Apresenta-se aqui o projeto de um sistema computacional para aquisição e análise de sinais de tensão e corrente da rede elétrica, utilizando a decomposição em Séries de Fourier. O sistema desenvolvido compreende uma placa de aquisição de dados ligada a um microcomputador, e um programa que faz o controle da placa e a análise dos dados.

Palavras-Chave: análise harmônica, interface, aquisição de dados.

1 Introdução

Equipamentos como computadores e motores de corrente contínua absorvem corrente elétrica da rede de alimentação de forma distorcida. Isto é, a corrente absorvida não é senoidal como deveria ser. Desta forma, são provocados vários danos à rede de fornecimento de energia elétrica, principalmente perda de energia no sistema de transmissão e distribuição.

O problema da má qualidade da energia elétrica e a sua escassez é atual, e não será resolvido a curto prazo, o que indica ser este um campo de pesquisa necessário para o momento e para o futuro. Para se ter um melhor aproveitamento da energia elétrica, evitando desperdício e outros danos ao sistema, torna-se imperiosa a adoção de medidas regulamentadoras do consumo de energia. No Brasil é exigido que os centros consumidores operem com fator de potência igual ou superior a 0,92 [Giacomini (1999)]. No exterior existem normas para regulamentação do consumo de energia por equipamentos eletrônicos, como a IEC 61000-3-2 (da CENELEC) e a IEEE 519-1981 (do Institute of Electrical and Electronics Engineers) [Sebástian (1997)].

Além de se criar normas para regulamentar o consumo de energia elétrica, é necessário investir em equipamentos de medição para verificação e fiscalização dos consumidores. Neste sentido se faz justo investigar formas de medição e análise do consumo de energia elétrica mais adequadas que as convencionais.

Este artigo tem por objetivo apresentar um equipamento que realize a aquisição de sinais elétricos, e um programa que controla a interface e analisa os dados obtidos.

2 O Hardware

A Figura 1 apresenta o diagrama em blocos das principais partes do hardware desenvolvido. Ao microcomputador é ligada uma interface paralela baseada no circuito integrado 8255 [Harris (1998)]. Essa interface pode executar as seguintes funções: enviar sinais de controle ao conversor analógico/digital (A/D), enviar dados para o conversor digital/analógico (D/A), receber o sinal de temporização, e receber dados do conversor A/D. A seguir tem-se uma melhor descrição dos componentes, e a programação do circuito.

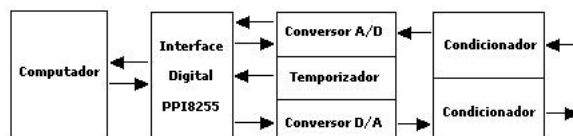


Figura 1: Diagrama da interface.

2.1 Descrição dos Componentes

Interface: Componente responsável pelo controle dos outros circuitos. Ela está ligada ao barramento de expansão ISA do computador [Eggebrecht (1990)], e é composta pelas seguintes partes:

- **Decodificador de Endereço:** Permite que a interface seja acessada em 4 endereços-base diferentes: 0300H, 0304H, 0308H ou 030CH (H indica base Hexadecimal). Os possíveis endereços-base estão separados de

4 unidades pois são necessários 4 endereços para operar a interface: um para cada porta A, B e C e um para a programação da interface.

- **82C55A:** Interface paralela programável (PPI) de 8 bits de comunicação com o microcomputador e 24 bits de comunicação com o exterior. Os 24 bits são agrupados em 3 portas de 8 bits cada: porta A, porta B e porta C; as quais podem ser programadas para funcionar como entrada ou saída, independentemente. A porta C pode ser subdividida em 2 partes, *C-low* e *C-high*, de 4 bits cada.

Temporizador: Este circuito é responsável por determinar o intervalo de aquisição de dados do circuito de amostra e retenção do conversor A/D. É importante que as amostras sejam feitas a intervalos bem regulares para que a análise forneça resultados precisos.

No Brasil, a rede elétrica opera com frequência de 60Hz, o que resulta em períodos de 16,667ms. As normas que cuidam da introdução de harmônicos na rede elétrica indicam a necessidade de se estudar até o 40º harmônico [Sebastião (1997)], ou seja, $40 \times 60\text{Hz}$, o que resulta em um período máximo de 0,417ms. Segundo a teoria da amostragem, é necessário que se tenha pelo menos 2 amostras do sinal de maior frequência a ser estudada [Hait (1975)]. Portanto chega-se a um intervalo máximo de amostragem de 0,209ms. Então foi estabelecido um valor mais exato para o período de amostragem, 0,200ms, ou seja, 83 amostras dos sinais de tensão e corrente da rede elétrica a cada período.

Para se conseguir intervalos regulares, foi construído um circuito baseado em um oscilador de cristal, com frequência de 10,000MHz, e um divisor digital de frequência.

Condicionadores: São usados para ajustar os sinais de tensão e corrente do meio externo aos valores de tensão entre 0V e 5V, proporcionalmente. São necessários para um bom funcionamento dos conversores.

Conversores: Foi utilizado o conversor A/D ADC0804 [National (1998)], que recebe sinais de controle de leitura e escrita provenientes da interface paralela (8255). Existe um circuito de amostra e retenção para o conversor A/D, com a função de manter constante o valor do sinal amostrado durante todo o período de conversão.

O conversor A/D recebe um valor de tensão entre 0V e 5V e fornece para a interface um número de 8 bits, proporcional ao valor recebido (variando entre 0 e 255).

O conversor D/A DAC0801 [National (1995)] recebe um valor de 8 bits e fornece uma tensão proporcional a este, que varia entre 0V e 5V. O seu tempo de conversão é de $2\mu\text{s}$.

2.2 Programação da Interface

Neste trabalho, a porta A e a Porta *C-high* foram programadas como entradas, e as portas B e *C-low* foram programadas como saídas, com as seguintes funções (os endereços das portas na primeira configuração da placa estão entre parênteses. Outros endereços possíveis podem ser obtidos somando esses valores com 4H, 8H ou CH):

Porta A: (0300H) envia dos dados ao conversor D/A.

Porta B: (0301H) recebe os dados do conversor A/D.

Porta C-Low: (0302H) 2 bits de saída, um para habilitar nova conversão (indicando que os dados foram lidos com sucesso), o outro indica que deseja-se ler o resultado da conversão, o que significa que o conversor A/D deverá colocar o valor convertido na porta B.

Porta C-High: (0302H) utiliza apenas um bit de entrada, para receber o sinal de tempo.

Configuração da Interface: (0303H) permite a programação das portas do circuito.

Os valores dos sinais de controle para a Porta C estão na Tabela 1.

Tabela 1: Valores para a Porta C.

Nome	Valor
RD	01H
WR	02H
INTR	10H
RESET	RD
CONVERTE	RD WR
LEITURA	WR

3 O Software

Desenvolveu-se um programa gráfico multiplataforma, intitulado Harmônicos, para o controle da interface. Esse foi desenvolvido em C++ [Stroustrup (1996)],

com alguns trechos em *assembly*, e utilizando os recursos da orientação a eventos, fornecido pela biblioteca gráfica *wxWindows* [Roebing (2000)].

3.1 Aquisição do Sinal

Uma nova aquisição é feita ativando a opção Nova Aquisição. Então aparecerá a tela mostrada na Figura 2.

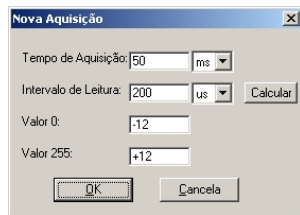


Figura 2: Nova aquisição.

O usuário deve escolher o tempo de aquisição e os valores “mínimo” e “máximo” da aquisição, para que o programa possa converter os dados de 8 bits em valores reais, como -12 e 12 , na Figura 2. O “intervalo de leitura” foi utilizado caso o programa de aquisição admita essa possibilidade, mas não é utilizado atualmente.

Quando o usuário pressiona o botão OK, um segundo programa é chamado, em segundo plano, para fazer as aquisições.

3.1.1 Programa de Aquisição

Este programa, também em C++, recebe como entrada o número de aquisições a serem efetuadas. Ele executa os seguintes passos:

- 1 Configura Placa
- 2 **Para** i **De** 1 **Até** $num_aquisicoes$
- 3 $v[i] := nova_aquisicao()$
- 4 **Fim Para**
- 5 **Para** i **De** 1 **Até** $num_aquisicoes$
- 6 $imprime_resultado(v[i])$
- 7 **Fim Para**

A função para imprimir os dados é a própria saída padrão (*cout*), que é capturada pelo programa Harmônicos, com um valor por linha.

As duas iterações poderiam ser agrupadas, diminuindo o número de comparações em *num_aquisicoes*, mas foi observado que o tempo de processamento necessário para imprimir cada número é maior que o tempo de várias aquisições, causando perda de dados.

A seguir está o trecho de código para as funções de leitura (*inportb*) e escrita (*outportb*) na placa:

```
#define ushort unsigned short
#define uchar unsigned char

static inline int inportb(int port)
{
    uchar value;
    asm volatile ("inb %1,%0"
                  : "=a" (value)
                  : "d" ((ushort) port));
    return value;
}

static inline void outportb(ushort port,
                             uchar value)
{
    asm volatile("outb %0,%1\n"
                : : "a" (value), "d" (port));
}
```

A função para fazer uma aquisição está descrita abaixo, em pseudo-código:

- 1 envia sinal de RESET para PORTA_C
- 2 envia sinal de CONVERTE para PORTA_C
- 3
- 4 **Fazer**
- 5 $v := ler\ sinal\ de\ PORTA_C$
- 6 **Enquanto** bit INTR de v estiver desativado
- 7
- 8 envia sinal de LEITURA para PORTA_C
- 9 ler sinal de PORTA_B
- 10 **Retornar**

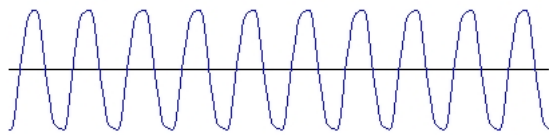
O primeiro passo do algoritmo é a limpeza do buffer de saída do conversor A/D, para iniciar uma nova conversão. A iteração verifica o final da conversão (ativação de INTR), quando a leitura poderá ser habilitada e então executada.

Pelo fato de chamar outro programa para fazer aquisição, o programa Harmônicos pode ser utilizado como interface para qualquer outro programa de aquisição de dados, desde que sejam satisfeitos os pré-requisitos de manipulação de dados.

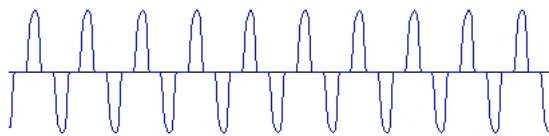
3.1.2 Resultados das Aquisições

A maioria das aquisições feitas foram bem sucedidas, mas ocorreram problemas em alguns testes, por causa do escalonamento de processos do sistema operacional (Windows NT), havendo perda de valores durante a aquisição.

Na Figura 3 estão as formas de onda de tensão e corrente adquiridas de um circuito retificador eletrônico com filtro capacitivo, que é conhecido popularmente como eliminador de pilhas.



(a) Forma de onda da tensão.



(b) Forma de onda da corrente.

Figura 3: Formas de onda de um retificador eletrônico com filtro capacitivo.

O programa é capaz de salvar os dados obtidos em um arquivo de pontos, de formato texto, além de poder carregar arquivos nesse formato, para sua análise.

3.2 Análise do Sinal

Para analisar os dados é necessário ter uma onda carregada pelo programa. O usuário deve então escolher as opções da análise (período da onda e número de harmônicos a serem analisados), para então o programa analisar e expor os resultados.

A análise dos dados está dividida em duas partes: a decomposição dos dados através da Série de Fourier, e o cálculo da Taxa de Distorção Harmônica.

3.2.1 Séries de Fourier

A decomposição em série de Fourier [Hait (1975)] é um importante método analítico que permite representar a maioria das funções matemáticas que se repetem f_o vezes por segundo. Estas funções são descritas pela soma de um número infinito de senos e cossenos, com frequências que sejam múltiplos inteiros de f_o , chamados de componentes harmônicos.

Todo sinal periódico pode então ser descrito através da seguinte fórmula:

$$f(x) = \sum_{k=0}^{\infty} a_k \cos(k\omega_0 x) + b_k \sin(k\omega_0 x),$$

com $\omega_0 = 2\pi f_o$.

Nota-se que, para $k = 0$, a função cossenoidal terá valor sempre 1, e a função senoidal terá sempre valor 0. Então a expressão fica da seguinte forma:

$$f(x) = a_0 + \sum_{k=1}^{\infty} a_k \cos(k\omega_0 x) + \sum_{k=1}^{\infty} b_k \sin(k\omega_0 x).$$

É demonstrado em [Hait (1975)] que os coeficientes a_k e b_k são calculados como:

$$a_k = \frac{2}{T} \int_0^T f(t) \cos(k\omega_0 t) dt$$

e

$$b_k = \frac{2}{T} \int_0^T f(t) \sin(k\omega_0 t) dt,$$

com

$$T = \frac{1}{f_0}$$

Neste trabalho, por não se conhecer $f(t)$, mas apenas uma sequência de pontos adquiridos do meio externo, a análise é feita através da integral por pontos. Então, dado um vetor P com n pontos (aquisições), o cálculo da integral é feito da seguinte forma:

$$a_0 = \frac{1}{n} \sum_{i=0}^n P[i],$$

$$a_k = \frac{2}{n} \sum_{i=1}^n P[i] \cos(k\omega_0 i),$$

e

$$b_k = \frac{2}{n} \sum_{i=1}^n P[i] \sin(k\omega_0 i).$$

O programa utiliza essas fórmulas para decompor as ondas. Outra forma de descrever a Série de Fourier é através de amplitudes (c_k), cossenos e ângulos de defasagem (θ_k), pois a soma de senos e cossenos de mesma frequência pode ser expressada dessa maneira:

$$f(x) = \sum_{k=0}^{\infty} c_k \cos(k\omega_0 x + \theta_k).$$

Os valores de c_k e θ_k podem ser calculados em função de a_k e b_k :

$$c_k = \sqrt{a_k^2 + b_k^2} \quad \theta_k = \tan^{-1} \left(\frac{b_k}{a_k} \right)$$

Essa forma é utilizada pelo programa para mostrar os resultados.

3.2.2 Taxa de Distorção Harmônica – TDH

A TDH da corrente elétrica indica o quão distorcida está a onda de corrente presente na rede de alimentação elétrica:

$$\text{TDH} = \frac{\sqrt{\sigma}}{c_0}, \text{ onde } \sigma = \sum_{i=1}^n c_i^2$$

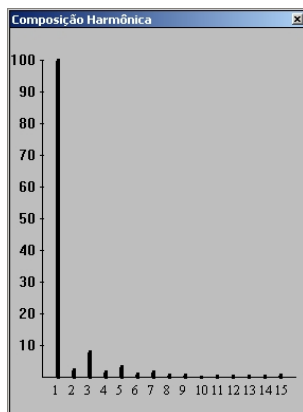
A TDH está diretamente relacionada ao fator de potência, conforme abaixo:

$$FP = \frac{\cos(\phi_1)}{\sqrt{1 + TDH^2}},$$

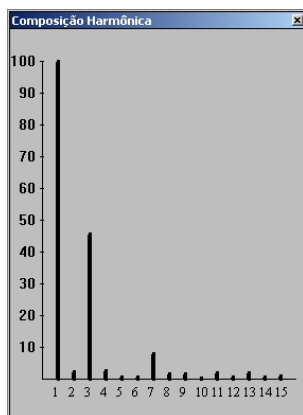
onde ϕ_1 é o ângulo de defasagem entre as componentes fundamentais (1º harmônico) de tensão e corrente.

3.2.3 Resultados dos Experimentos

Foram analisadas as formas de onda da Figura 3. Na Tabela 2 estão os resultados numéricos das análises. Na Figura 4(a) está o resultado gráfico da análise da forma de onda da Figura 3(a), com TDH 0,223. Na Figura 4(b) está o resultado da análise da forma de onda da Figura 3(b), com TDH 0,5.



(a) Gráfico da análise da tensão.



(b) Gráfico da análise da corrente.

Figura 4: Resultado das análises.

Nota-se que a análise da forma de onda de corrente apresentou uma grande incidência de 3º e 7º harmônicos, os quais provocam distorções na onda de tensão, que devem ser evitadas.

Tabela 2: Valores das análises de tensão e corrente.

Harmônico	Tensão		Corrente	
	Valor	Ângulo	Valor	Ângulo
1	258,283	136,268	174,966	135,700
2	6,373	55,571	4,036	32,531
3	20,774	146,701	79,913	46,791
4	4,316	79,415	4,930	130,341
5	8,589	174,798	0,902	54,870
6	2,492	107,426	1,446	132,223
7	3,969	109,334	14,145	35,334
8	1,436	148,944	2,659	163,314
9	2,123	132,245	2,637	72,283
10	0,135	42,276	0,536	32,921
11	1,141	114,222	3,651	5,811
12	0,516	49,549	1,173	73,212
13	0,870	64,471	3,208	87,014
14	0,866	104,366	1,444	78,946
15	1,489	107,994	1,736	91,083

Os dados analisados podem ser salvos pelo programa, em um arquivo formato texto, já formatado em colunas.

3.2.4 Recomposição do Sinal

Os sinais de onda foram recompostos com o programa Visugraf [Andrade (2002)], e os resultados das recomposições podem ser visualizados nas Figuras 5(a) e 5(b). Nota-se que ocorreram alguns erros na recomposição das ondas, que estão mais evidentes nas regiões onde a corrente se manteve constantemente nula. Esses erros são atribuídos a problemas da precisão numérica utilizada nos cálculos.

4 Conclusões

Um dos principais motivos para desenvolvimento deste projeto foi a verificação da integração *hardware/software*, e a geração de um programa de controle eficiente para o sistema integrado.

Ocorreram problemas em algumas aquisições, devido ao escalonamento de processos do sistema operacional, havendo perda de valores durante a aquisição. Se o sistema for utilizado para fazer aquisições com intervalos de tempo maiores (10 ou menos aquisições por segundo) estes problemas não são percebidos.

Como projetos futuros está resolver o problema das perdas nas aquisições. Como exemplos pode-se citar:

- fazer aquisições utilizando a entrada de áudio da placa de som do computador, pois ela tem um buffer interno, que pode acumular dados enquanto o sistema não faz o tratamento;

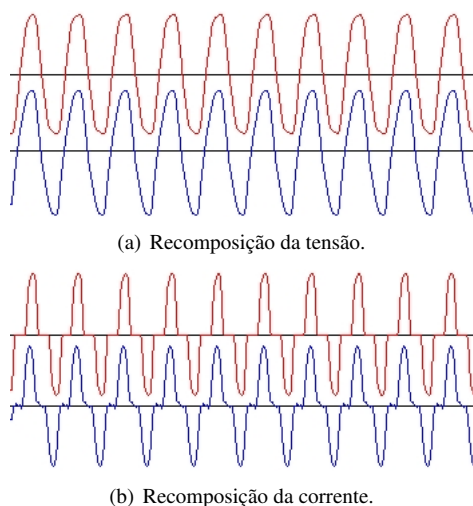


Figura 5: As ondas e suas recomposições: as mais acima são as originais, as mais abaixo são as recompostas.

- a criação de uma interface dedicada, com um microcontrolador, para fazer toda a aquisição e depois apenas devolver o resultado para o computador;
- utilizar um sistema operacional em tempo real, como por exemplo o Linux com kernel real time.

Referências

- [Andrade (2002)] Andrade Neto, P. R. *Análise de Sinais Elétricos por Microcomputador*, 2002. URL: www.comp.ufla.br/~pedron/analise/.
- [Eggebrecht (1990)] Eggebrecht, L. C. *Interfacing to the IBM Personal Computer*, SAMS, 1990.
- [Giacomin (1999)] Giacomin, J. C. et alli. *Two Switching DC Power Supplies with High Power Factor Input: a Comparative Study*. Foz do Iguaçu, 5^o Congresso Brasileiro de Eletrônica de Potência, 563-568, 1999.
- [Hait (1975)] Hait Jr., W. H. & Kemmerly J. E. *Análise de Circuitos em Engenharia*, McGrawHill – São Paulo, 1975.
- [Harris (1998)] 82C55A – *CMOS Programmable Peripheral Interface*, Harris Semiconductor, 1998.
- [National (1995)] DAC0801 – *8-bit Digital-to-Analog Converters*, National Semi-

conductors Corporation, 1995. URL: <http://www.national.com/>.

[National (1998)] ADC0804 – *8-bit mP Compatible Converters*, National Semiconductors Corporation, 1998. URL: <http://www.national.com/>.

[Roebbling (2000)] Roebbling, R. et al. *wxWindows, Cross-Platform Development for Unix/Windows/MacOS*, 2000. URL: <http://www.wxwindows.org/>.

[Sebastião (1997)] Javier Sebastián et alli. *Power Factor Correction in Single Phase Switching Power Supplies*. Belo Horizonte, 4^o Congresso Brasileiro de Eletrônica de Potência, 14-28, 1997.

[Stroustrup (1996)] Stroustrup, B. *C++ a Linguagem de Programação*, Bookman – Porto Alegre, 3^a ed. 823p.