

Desenvolvimento de uma Interface de Aquisição e Controle de Dados

EDNA MIE KANAZAWA
WILIAN SOARES LACERDA

UFLA – Universidade Federal de Lavras
DEX – Departamento de Ciências Exatas
Cx Postal 37 – CEP 37200-000 Lavras (MG)

kanazawa@comp.ufla.br
lacerda@ufla.br

Resumo: Apresentamos neste trabalho uma Interface de Aquisição e Controle de Dados, utilizando um conversor analógico/digital da National Semiconductor (ADC 0804), um conversor digital/analógico (DAC 0801), um timer programável (8254) e um registrador de saída (74LS374). O sistema é conectado ao barramento de expansão do PC, e foi desenvolvido para fins didáticos e de pesquisa para o curso de graduação de Ciência da Computação.

Palavras Chaves: Interface, conversor A/D e D/A, timer, aquisição

1 Introdução

Interfaces são utilizadas para comunicação entre dois sistemas que possuem diferentes ambientes. Um computador digital processa apenas dados digitais (sinais discretos). Entretanto, o ambiente externo ao computador possuem sinais geralmente no formato analógico (sinais contínuos). Para que as informações coletadas sejam inseridas no computador então é necessário um tipo de interface que transforme os sinais analógicos em dados digitais a serem processados. Da mesma forma, para o computador interagir com o sistema externo, é necessário transformar os dados digitais em sinais analógicos compatíveis. O objetivo deste projeto é justamente descrever o desenvolvimento de uma interface capaz de interfacear um computador tipo PC [3] com um sistema externo qualquer.

A Interface de Aquisição e Controle de Dados proposta, possui as seguintes funções:

- converter um sinal analógico em um sinal digital;
- converter um sinal digital em um sinal analógico;
- gerar quatro bits de saída que podem controlar circuitos digitais externos;
- contagem de eventos externos, repetitivos ou não.

Existem diversas aplicações para Interface de Aquisição e Controle de Dados, tais como:

- monitoração e controle de temperatura de uma estufa;
- gerador de forma de ondas;
- medição de frequência de eventos repetitivos;
- contagem de eventos;
- monitoração e controle de luminosidade;
- monitoração e controle de vazão;
- monitoração e controle de velocidade de um motor, etc.

A Interface de Aquisição e Controle de Dados proposta tem ainda as seguintes características:

- simples;
- fácil utilização;
- baixo custo.

2 Funcionamento

Na Figura 1, é apresentado a Interface de Aquisição de Dados em diagrama em blocos. O decodificador é responsável pelo mapeamento das portas de I/O da interface no barramento de endereço do PC. O conversor digital-analógico converte o dado digital em um nível de tensão de saída. O conversor analógico-digital converte um nível de tensão de entrada em um dado digital de 8 bits. O circuito do timer é usado para o início automático da conversão A/D e para a

contagem de eventos externos. O registrador de saída, além de permitir o controle da interface via software, permite o controle de um processo externo através de 4 bits de saída. A seguir é descrito com detalhes cada

bloco. Na figura 2, é apresentado o esquema eletrônico completo da interface.

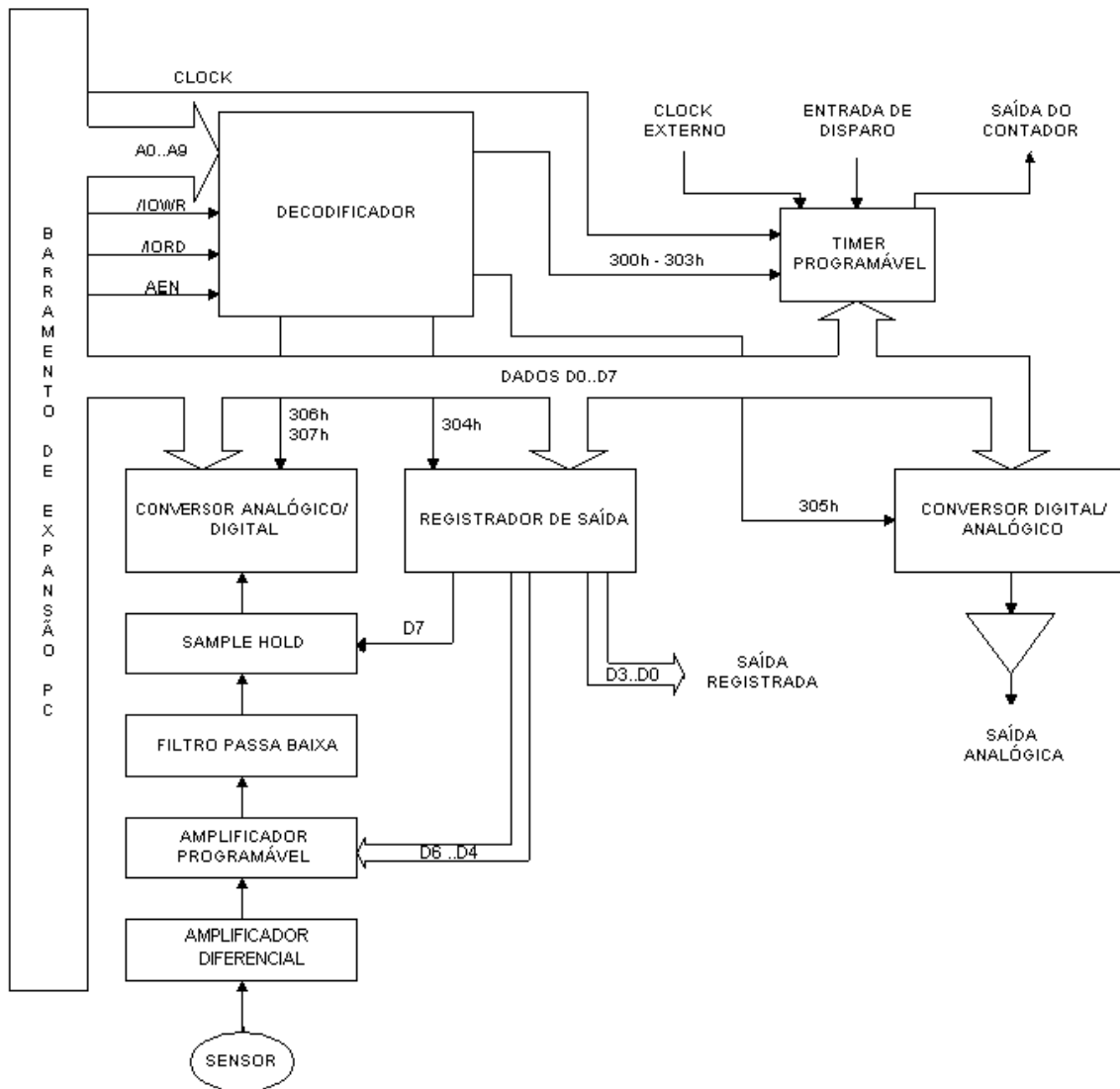


Figura 1: Diagrama em blocos da Interface de Aquisição e Controle de Dados

2.1 Decodificador

O circuito de decodificação utiliza as linhas de endereço (A0 a A9) e as linhas de controle (/IOWR, /IOR e AEN) do barramento do PC para decodificar os endereços das portas de I/O abaixo relacionadas:

- endereços 300h à 303h: habilitam o timer programável (8254) para leitura e escrita;

- endereço 304h: habilita o registrador (74LS374) para escrita;
- endereço 305h: habilita o conversor digital/analógico para escrita;
- endereço 306h: habilita o conversor analógico/digital para leitura;
- endereço 307h: habilita o início de conversão .

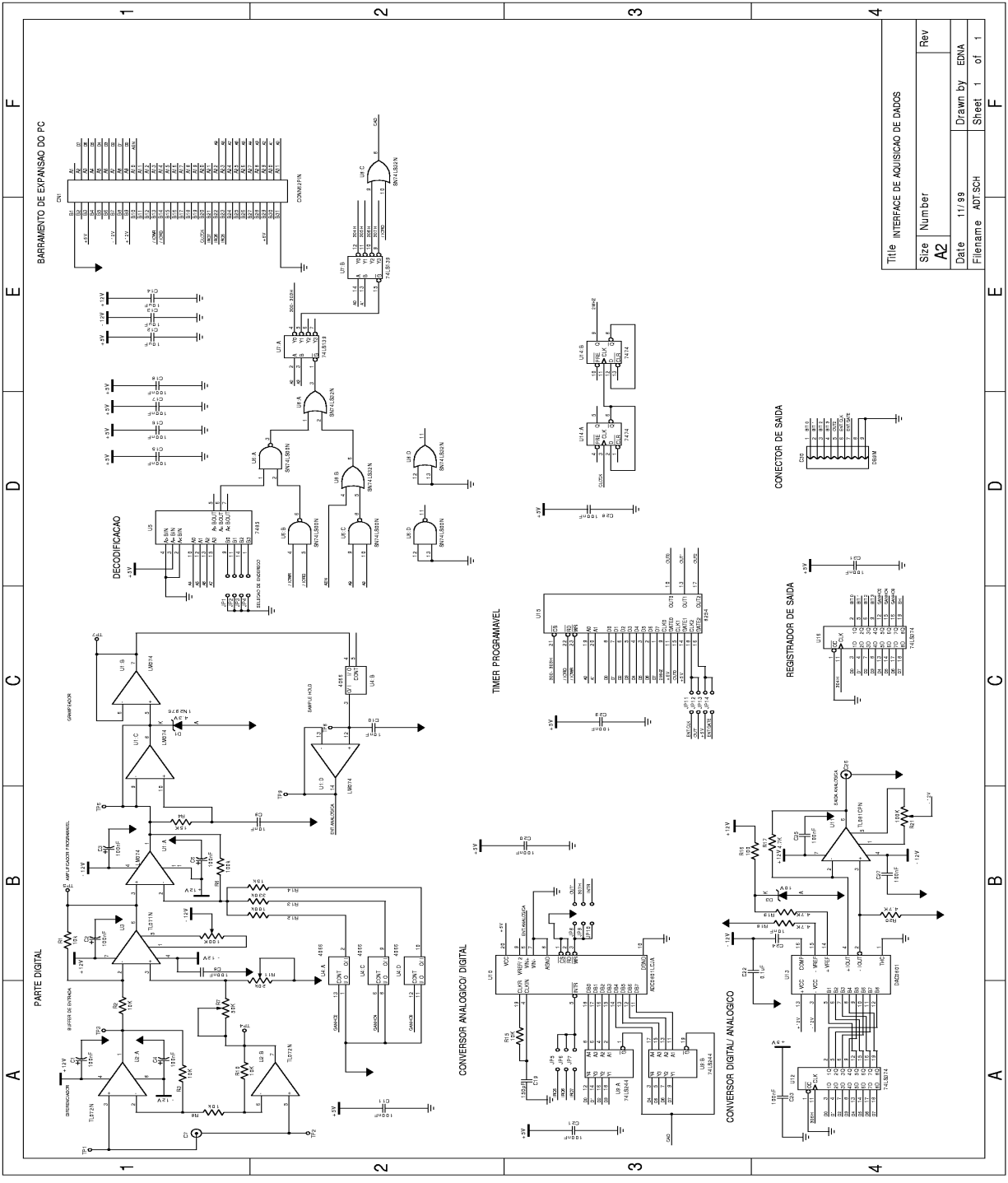


Figura 2: Esquema eletrônico da Interface de Aquisição e Controle de Dados

Title		INTERFACE DE AQUISICAO DE DADOS	
Size	Number	Rev	
A2			
Date	11/99	Drawn by	EDNA
Filename	ADT.SCH	Sheet	1 of 1

2.2 Conversor Digital/Analógico (DAC 0801)

O conversor DAC0801, converte um dado digital de 8 bits (D0 a D8), gerado via software, para um equivalente sinal analógico. A operação de conversão inicia-se quando é ativado o sinal de escrita, dado pela linha de controle /IOWR, no endereço 305h. Juntamente é enviado pelo barramento de dados a palavra a ser convertida.

O sinal analógico de saída do conversor pode variar de 10V à -10V de acordo com o valor digital escrito na porta.

2.3 Conversor Analógico/Digital (ADC 0804) com seus Circuitos Auxiliares

O conversor ADC0804 converte um sinal analógico em um correspondente dado digital de 8 bits (D0 a D8), que pode ser interpretado e processado via software. O nível de tensão da entrada analógica pode variar de 0 a 5V. O tempo de conversão é de 100us. A conversão pode iniciar de três formas: por endereçamento (escrita no endereço 307h), pela base de tempo do contador 1 do 8254, ou conversão contínua.

Alguns sensores [2, 4] podem ser acoplados à entrada analógica do conversor, por exemplo: sensor de temperatura (PTC, NTC, termopar), sensor de força (strain gage), sensores de posição (potenciômetro), etc. Circuitos auxiliares (amplificador, filtro passa-baixa, sample/hold) garantem a conformação do sinal proveniente do sensor para a aquisição.

2.3.1 Amplificador Diferencial

Sua função é rejeitar os sinais em modo comum, ou seja, atenuar qualquer ruído que surja no cabo de entrada do sinal. Seu ganho é unitário, entretanto, o ganho de malha positiva e negativa são ajustados independentemente, a fim de obter uma máxima rejeição em modo comum.

2.3.2 Amplificador Programável

Sua função é amplificar a tensão de entrada do sensor, já com o ruído em modo comum eliminado pelo amplificador diferencial. O ganho do amplificador é programável via software pelos bits D4, D5 e D6 do registrador de saída, que é acessado pelo endereço 304h.

2.3.3 Filtro Passa-Baixa

Sua função é permitir a passagem das componentes do sinal de entrada com frequências de até 1kHz,

evitando assim o efeito “aliasing” provocado pela amostragem do sinal analógico.

2.3.4 Sample/Hold

Permite a retenção do sinal de entrada analógico durante o processo de conversão. É controlado pelo bit D7 do registrador de saída.

2.4 Timer Programável (8254)

É responsável pela geração do sinal automático de início de conversão A/D. É controlado e programado por software, permitindo ao usuário especificar o período de conversão. Além disso o timer pode ser programado para contar a quantidade de pulsos de um sinal digital de entrada. Abaixo são citadas algumas aplicações de timer programável:

- gerador de clock em tempo real;
- contador de eventos;
- disparador digital;
- gerador de onda quadrada, etc.

O timer é programado via software, utilizando-se os endereços 300h, 301h, 302h, 303h.

No endereço 303h é gravada a palavra de controle, onde está especificado qual contador está sendo programado e o modo de programação.

2.5 Registrador de Saída (74LS374)

O registrador de saída é acessado pelo endereço 304h, e é usado para armazenar o dado proveniente do barramento de extensão. O dado controla: o sample/hold (1 bit), o ganho do amplificador programável (3 bits), e 4 bits de saída para uso externo.

3 Programação

A interface é controlada e programada via software, desenvolvendo em linguagem C++ [1]. A seguir será mostrado alguns exemplos de programação e aplicações para a interface.

3.1 Aplicação do Conversor A/D

Este programa exemplo mostra a utilização do conversor A/D para ler um sinal gerado por um sensor de temperatura (NTC), utilizando o compilador g++ 2.8.1.

```
#include <iostream>
#include <cstdlib>
#include <time.h>
//Função de leitura em porta
```

```

//port = numero da porta de entrada da
//interface
static inline int inportb( int port ){
    unsigned char value;
    asm volatile ("inb %1,%0"
                 : "=a"(value)
                 : "d"((unsigned short)port));
    return value;
}
//Função de escrita em porta
//port=número da porta de saída do computador
//val=byte a ser escrito na porta port
static inline void outportb( unsigned short int
    port, unsigned char val ){
    asm volatile ("outb %0,%1\n"
                 : : "a" (val), "d" (port));
}
//Função que gera a palavra do registrador de
//saída
//D0 a D3 correspondem aos bits de saída
//digital
//D4 a D6 correspondem a seleção de ganhos
//D7 corresponde ao sample/hold
//a palavra gerada por esta função é enviada
//a porta 304h
void registrador (int ganho, int samplehold,
                 int bitsaida){
    unsigned char palavra_nova;
    int palavra = (samplehold * 128 + ganho * 8 +
                 bitsaida);
    palavranova = palavra;
    outportb (0x304, palavranova);
}
int main (void){
    //ganhos do amplificador de entrada
    int ganho[7]={2,4,6,8,10,12,14};
    int bitsaida = 3;//bits de saída
    int entanalogica; // entrada analógica
    int i = 0;
    do{
        //seleciona o ganho
        registrador (ganho[i], 0, bitsaida);
        //fecha o sample/hold
        registrador (ganho[i],1,bitsaida);
        // tempo para carregar o sample/hold
        sleep (2000);
        //abre o sample/hold
        registrador (ganho[i],0,bitsaida);
        //inicia a conversão
        outportb (0x307,0x00);
        //tempo de conversão
        sleep (2000);
        //faz a leitura do sinal analógico
        entanalogica = inportb(0x306);

```

```

//aumenta o ganho
    i++;
}while ((entanalogica <= 127) && (i < 6));
double resultado = entanalogica *
    0.0196/ganho[i-1];
int temperatura = (int)(resultado * (-30) +
    99);
cout<<"temperatura:" << temperatura << endl;
system ("pause");
return 1;
}

```

3.2 Programação do 8254

O programa abaixo descreve como programar o timer para gerar uma forma de onda quadrada de saída na saída do contador 2.

```

int main(void){
    system("cls");
    cout << "\n****Programacao do 8254****\n";
    //programando contador 0
    outportb(0x303,0x16);
    //grava palavra de contagem contador 0
    outportb(0x300,0xff);
    //programando contador 1
    outportb(0x303,0x56);
    //grava palavra de contagem contador 1
    outportb(0x301,0x0f);
    //programando contador 2
    outportb(0x303,0x96);
    //grava palavra de contagem contador 2
    outportb(0x302,0x0f);
    return 0;
}

```

3.3 Aplicação do Conversor D/A

O programa abaixo mostra a utilização do conversor D/A para gerar uma forma de onda dente de serra.

```

int main (void){
    int nivel_0 = 128; //nível 0
    int max_ampl = 10;
    int numero;
    int amplonda = 0;
    for (amplonda = -10 ; ; amplonda++){
        numero = amplonda * nivel_0/max_ampl +
            nivel_0;
        outportb(0x305, numero);
        sleep(50);
        if (amplonda == 10) amplonda = -10;
    }
    return 0;
}

```

4 Conclusões

A Interface de Aquisição e Controle de Dados atende as expectativas esperadas. Através da interface é possível facilmente realizar aquisição de dados proveniente de sensores, e controlar um processo externo qualquer via software.

A interface desenvolvida atende uma ampla gama de aplicações, mantendo sua característica de simplicidade e baixo custo.

5 Referências

- [1] DEITEL, H. M. & DEITEL, P. J. *C++ How to program*. 2^a ed. Upper Saddle River, New Jersey: 1998. 1130 p.
- [2] DOEBELIN, Ernest O. *Measurement System, Application and Design*. 4^a ed. Singapore: 1990. 960 p.
- [3] EGGBRECHT, Lewis C. *Interfacing to the IBM personal computer*. 2^a ed. Estados Unidos: 1990. 345 p.
- [4] HELFRICK, Albert D. & COOPER, Willian D. *Instrumentação Eletrônica Moderna e Técnicas de medição*. Rio de Janeiro: 1994. 324 p.