

Comunicação entre microcomputadores

Conheça a comunicação no IBM PC usando as interrupções

Wladimir Lopes Silva

Quando ocorre uma interrupção em um microcomputador padrão IBM-PC, os flags e registradores são colocados em uma pilha. Neste momento os registradores são carregados com novos valores contidos no vetor de interrupção, associado com o elemento do hardware que originou a interrupção. Isto significa que as próximas instruções a serem executadas pela CPU serão da rotina de tratamento da interrupção desse elemento de hardware. Esta rotina salva todos registradores que serão modificados, executa sua função, restaura os registradores anteriormente salvos e emite um IRET. A instrução IRET retira os registradores da pilha encerrando a interrupção. O programa que havia sido interrompido continua sendo executado sem sofrer nenhum efeito da interrupção.

Este modo de tratamento de interrupções pode ser utilizado para implementar uma comunicação de dados entre microcomputadores. Nos sistemas de comunicação utilizando interrupção, a rotina de tratamento de interrupção é o processo-produtor que recebe caracteres da porta de entrada e os coloca em um buffer, a partir deste momento podem ser utilizados pelo

processo-consumidor para serem tratados e processados conforme a necessidade e o objetivo do sistema em questão.

O tamanho dessa rotina de tratamento de interrupção deve ser mínimo para permitir altas taxas de recepção/transmissão. O dimensionamento do buffer de recepção de dados é função do tempo de processamento do processo-consumidor, ou seja, quanto mais vagaroso for o processamento, maior deve ser o buffer de recepção, para que não haja perda de informação.

O hardware envolvido neste processo consiste do Controlador Programável de Interrupções (8259), da UART (8250 ou 16450 - Universal Asynchronous Receiver/Transmitter), além da CPU.

O chip 8259 é responsável pelo controle das interrupções e a UART pelo suporte a comunicação serial.

Estes dispositivos serão detalhados durante o texto. Pretende-se que após a leitura atenta deste texto, os programadores interessados em desenvolvimento de sistemas tenham os subsídios necessários para desenvolver programas utilizando o hardware da porta serial.

CONTROLADOR PROGRAMÁVEL DE INTERRUPTÕES - 8259

Os computadores padrão IBM-PC possuem vários tipos de interrupções de hardware que podem ocorrer simultaneamente. O chip 8259 da INTEL, também conhecido como PIC (Programmable Interrupt Controller) é responsável pelo controle destas interrupções, definindo a execução de cada uma delas.

O 8259 recebe um sinal do hardware solicitando a execução de uma interrupção. Caso ocorra uma outra solicitação ao mesmo tempo, ele verifica qual é a prioritária e passa o controle para a CPU, que se encarrega de executar a rotina associada ao dispositivo. A execução de menor prioridade será executada assim que o 8259 puder liberá-la.

Existem 8 níveis de prioridades de interrupções no IBM-PC XT e 16 ou mais níveis de prioridade nos modelos tipo IBM-PC AT. Estes níveis encontram-se numerados de 0 a 7, nos micros XT e de 0 a 15 nos microcomputadores AT e são denominados IRQ (Interrupt Request - Requisição de Interrupção).

A tabela 1 (na página ao lado) ilustra a ordem de prioridade de cada interrupção de hardware controlada pelo 8259.

IRQ (PRIORIDADE)	INTERRUPÇÃO	OBSERVAÇÃO
0	Relógio	
1	Teclado	
2	Canal de Entrada e Saída	
3	Porta Serial COM1: (COM2: no IBM-PC AT)	Vetor de Interrupção 11 (0Bh)
4	Porta Serial COM2: (COM1: no IBM-PC AT)	Vetor de Interrupção 12 (0Ch)
5	Disco Rígido	
6	Controlador de Disco Flexível	
7	Porta Paralela LPT1: (Impressora)	
8 *	Relógio de Tempo Real	
9 *	Redirecionamento para IRQ2	
10	Reservada	
11	Reservada	
12	Reservada	
13 *	Co-processador Matemático	
14 *	Controlador de Disco Rígido	
15	Reservada	

* Somente nos modelos IBM-PC AT

TABELA 1

ASSISTÊNCIA TÉCNICA PERFEITA

MICROLOGICA PORQUE ?

Temos 7 anos de atuação no mercado, realizando serviços, utilizando engenheiros especialistas em reparos de micros PC, XT, AT, 386, 486, Drives, Impressoras, Monitores e outros periféricos.



COMPROVE!!!

DEFEITOS EM MICROCOMPUTADORES: Erros de lógica aleatórios, perda de memória, destruição do software, falha de componentes...
UMA SOLUÇÃO PERFEITA: MICROLÓGICA, o melhor caminho para eliminar defeitos em microcomputadores e periféricos.

ALUGUE MICRO DE 16, 32 BITS

CONTRATOS DE ASSISTENCIA TECNICA



MICROLOGICA TUDO PARA INFORMATICA

Engenharia de Sistemas Ltda. Consultoria de Hardware

RUA CAMERINO, 128 - 11º ANDAR - CENTRO - RIO (PRÓXIMO A EST. DO METRÔ PRES. VARGAS)

Tel.: (021) 263-9925 / 263-9408 / 233-6826

As interrupções são identificadas por valores de 8 bits (0 a 255) quando passadas para a CPU. Para cada interrupção solicitada, associa-se um endereço de memória onde inicia a rotina correspondente de tratamento da mesma. Estes endereços estão armazenados no Vetor de Interrupções que inicia-se na posição de memória 0000:0000.

Cada um destes endereços é armazenado em 4 bytes, 2 bytes para o deslocamento e 2 bytes para o segmento.

A CPU do microcomputador comunica-se com os controladores de dispositivos, como é o caso do 8259, através de portas lógicas de entrada e saída. Estas portas lógicas permitem passar e receber informações dos dispositivos, determinando seu estado ou verificando algum valor processado por ele.

O 8259 permite ser configurado através de portas. Quando se liga o computador, o BIOS (Basic Input Output System, se encarrega de configurá-lo). As portas do 8259 que interessam neste caso correspondem aos seguintes endereços:

- 20H - Registro de Comando de Interrupções (ICR);
- 21H - Registro de Máscara de Interrupções (IMR);

Através do IMR, porta 21h, pode-se habilitar uma interrupção de hardware. Os bits setados em 1 neste registro correspondem a interrupções mascaradas, isto é, interrupções não permitidas. Caso seja lido na porta 21h (IMR) o valor F0h (11110000 em binário), significa que as IRQ0, IRQ1, IRQ2, IRQ3 estão habilitadas, ou seja, podem ser executadas.

A porta 20h (ICR) permite passar comandos para o 8259. Neste caso será enviado, após a interrupção, o comando "END OF INTERRUPT" (EOI), indicando ao 8259 o final da mesma. Caso isto não ocorra, o 8259 fica aguardando o final efetivo da interrupção e não habilita outra que venha a ocorrer.

UART - UNIVERSAL ASYNCHRONOUS RECEIVER / TRANSMITER

A UART é responsável pela implementação da comunicação serial

nos microcomputadores padrão IBM. O chip utilizado depende do modelo de microcomputador em questão:

- modelo IBM-PC XT - UART 8250;
- modelo IBM-PC AT - UART 16450;
- modelo IBM-PS/2 - UARTs 16550 e 16550A.

Estes chips são de fabricação da National e acompanham a evolução dos novos modelos de microcomputadores.

A UART é um chip programável e apresenta vários registros internos acessados via software. A figura 1 (na página ao lado) mostra o diagrama de blocos da UART com seus principais registros. Estes registros definem o funcionamento da UART, inclusive transmissão e recepção de dados:

THR - Transmit Holding Register: é o registro da UART que contém o caractere ASCII a ser transmitido pela porta serial;

RBR - Receive Buffer Register: é o registro da UART que contém o caractere recebido pela porta serial;

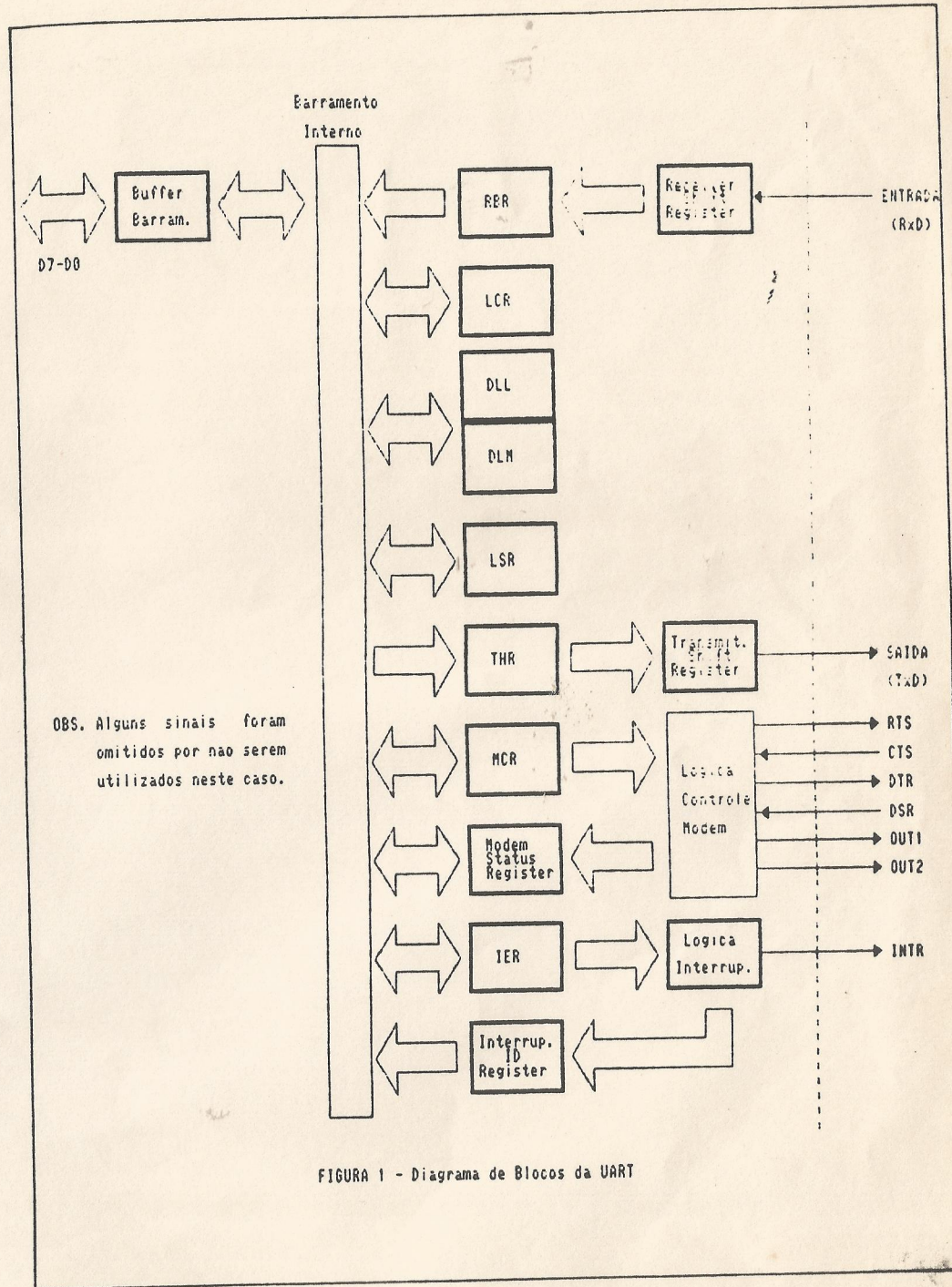


FIGURA 1 - Diagrama de Blocos da UART

RAISFER SHAREHOUSE TEL. 031-443-3020

A PRIMEIRA SHAREWARE DAS GERAIS - BELO HORIZONTE-MG

- EXCLUSIVO PARA PC XT E AT
- LANÇAMENTO SIMULTÂNEO COM EUROPA E U.S.A
- FAÇA SEU PEDIDO POR TELEFONE OU FAX
- APÓS ÀS 21:00 HORAS LIGUE A COBRAR
- ATENDEMOS ATÉ ÀS 24:00 HORAS
- SOLICITE CATÁLOGO GRATUITO
- REMETEMOS SEU PEDIDO EM MENOS DE 24HS
- PAGUE SOMENTE QUANDO RECEBER

**FAÇA JÁ O SEU PEDIDO !
NOVO ENDEREÇO
031-443-3020
AV. XANGRI-LA, 75 - C125 - BRAUNAS
BELO HORIZONTE - MG
CEP: 31.365-640**

PREÇO POR DISCO (INCLUSO)

5 1/4DD	CR\$ 37.900,00
5 1/4HD	CR\$ 72.200,00
3 1/2HD	CR\$ 89.900,00

OBS: PREÇOS VÁLIDOS ATÉ 30/04/93

LANÇAMENTOS		TETRIS CLASSIC	
INCA	10/HD	THE LEGEND OF KYRANDIA	01/HD
SPEAR OF DESTINY	02/HD	POLICE QUEST I - VGA	04/HD
ALONE IN THE DARK	04/HD	CHESS MASTER 3000	06/HD
BATMAN RETURNS	07/HD	ACES OF PACIFIC - MISSION	01/HD
WIEN THE PROPHECY	05/HD	ELVIRA II	03/HD
DRAGONS LAIR IV	05/HD	EYE OF BEHOLDER II	03/HD
ULTIMA UNDERWORLD II	05/HD	REX NEBULAR	10/HD
X-WING	05/HD	V FOR VICTORY	01/HD
BATTLECHESS 4000	08/HD	TASK FORCE 1942	08/HD
POPULOUS II	05/HD	DARKLANDS	11/HD
SPACE QUEST V	05/HD	HUMANS	01/HD
LURE OF TEMPTRESS	04/HD	COMANCHE MAX OVERKILL	03/HD
TERMINATOR 2029	07/HD	INDIANA JONES - ATLANTIS	06/HD
THE DARK HALF	04/DD	CREATERS	01/HD
RISK WOODS	01/HD	OUT OF THIS WORLD	01/HD
DAUGHTER OF THE SERPENTS	08/HD	GOBLINS II	01/DD
CURSE OF ENCHANTIA	04/HD	T.J.M. THE INCREDIBLE MACHINE	01/HD
		GLOBAL EFFECT	01/DD
		GUNSHIP 2000 MISSIONS	02/HD
F-15 STRIKE EAGLE III	06/HD	FIRST SAMURAI (VGA)	01/HD
DARK SEED	05/HD	AMAZON	06/HD
SHERLOCK HOLMES	10/HD	LINKS 386 PRO	04/HD
GREAT NAVAL BATTLES	03/HD	CAR AND DRIVER	06/HD
XF 5700 - MANTIS	09/HD	HEIMDALL	03/HD
PROPHECY OF SHADOW	03/HD	HOOK	03/HD
CRUSADERS DARK SAVANT	03/HD	SUMMER CHALLENGE	01/HD
GUY SPY	02/HD	THE DAGGER OF AMON RA	05/HD
FALCON 3.0 MISSIONS	02/HD	MONKEY ISLAND II	06/HD
DUNE II	04/HD	CRISIS IN THE KREMLIN	04/HD
QUEST FOR GLORY III	06/HD	ULTIMA 7 - BLACK GATE	06/HD
STUNT ISLAND	06/HD	ABC BOXING	01/HD
KING QUEST VI	08/HD	WORLD CIRCUIT - F1 GP	03/HD
PIEVEL AN AMERICAN TAIL	05/DD	WING COMMANDER 2	07/HD
SPACE ACE II	03/HD	GODS	01/HD
ATP AIR TRANSPORT	02/HD	KNOWLEDGE ADVENTURE	05/HD
HONG KONG MAJON PRO	02/HD	ISAAC ASIMOV'S ADVENTURE	05/HD
ANCIENT ART OF WAR	03/HD	E MUITO MAIS PARA VOCE!	

DLL - Divisor Latch LSB : é um dos registros internos da UART que permite definir a taxa de transmissão / recepção dos dados.

Taxa (Bauds) (Dec)	Divisor (Hex)	DLL (Hex)	DLM (Hex)
50	2304	= (00h)	09h
75	1536	= (00h)	06h
110	1047	= (17h)	04h
134,5	857	= 59h	03h
150	768	= 00h	03h
300	384	= 80h	01h
600	192	= C0h	00h
1200	96	= 60h	00h
1800	64	= 40h	00h
2000	58	= 3Ah	00h
2400	48	= 30h	00h
3600	32	= 20h	00h
4800	24	= 18h	00h
7200	16	= 10h	00h
9600	12	= 0Ch	00h
19200	6	= 06h	00h
38400	3	= 03h	00h
57600	2	= 02h	00h
115200	1	= 01h	00h

DLM - Divisor Latch MSB : é um dos registros internos da UART que permite definir a taxa de transmissão / recepção dos dados.

LCR - Line Control Register : é o registro de controle da UART, através dele são especificados o formato dos dados e o carregamento dos registros DLL e DLM que determinam a taxa de transmissão de dados.

Os bits 0 e 1 permitem selecionar o tamanho do caracter a ser transmitido ou recebido serialmente (5 / 6 / 7 / 8 bits). O bit 2 define o número de "stop bits" transmitido e recebido em cada caracter (1 / 1,5 / 2 "stop bits").

O bit 3 permite habilitar a paridade, ou seja, um bit de paridade é gerado em caso de transmissão ou verificado em caso de recepção de dados. A paridade é usada para checar os dados enviados e recebidos.

O bit 4 define se esta paridade será ímpar ou par. O bit 5 permite definir se o bit de paridade a ser gerado ou testado será função do número de bits no estado lógico 1 ou 0. A figura 2 (ao lado) mostra um exemplo de formato final do caracter enviado ou recebido serialmente.

O bit 6 não é utilizado nesta aplicação. O bit 7 (DLAB - Divisor Latch Access Bit) habilita o acesso aos registros internos da UART responsáveis pela definição da taxa de transmissão / recepção dos dados. A UART gera um sinal de saída cuja a frequência é função destes registros internos.

Os registros DLL e DLM são respectivamente o byte menos significativo e o byte mais significativo do divisor que define esta taxa. A UART permite as seguintes taxas de transmissão / recepção:

Portanto, a taxa de transmissão / recepção é dada pela equação:

$$\text{Taxa} = 115200 / \text{Divisor}$$

MCR - Modem Control Register : é o registro da UART que controla uma miscelânea de funções de modem. O bit 0 permite atuar no DTR (Data Terminal Ready), este sinal é necessário em alguns modems, neste caso o estado lógico 0 interrompe a comunicação. O bit 1 define o estado do sinal RTS (Request to Send), ele implementa um protocolo juntamente com o sinal do pino de entrada CTS em comunicações Half-Duplex. Em Full-Duplex este sinal permanece em estado lógico 1 durante toda comunicação. O bit 2 permite atuar na saída OUT1 da UART que está a disposição, algumas placas ou modems podem utilizá-la. O bit 3 define o estado da saída OUT2 da UART, este sinal é utilizado na arquitetura dos microcomputadores padrão IBM-PC para permitir a ocorrência de alguns interrupções. O bit 4 inicializa um teste

interno na UART e os bits 5, 6 e 7 não são utilizados.

IER - Interrupt Enable Register: este registro da UART permite habilitar os 4 tipos possíveis de interrupção. Será utilizado apenas o bit 0 que habilita a interrupção quando o dado recebido está pronto para ser usado pelo sistema.

Alguns registros internos da UART não foram detalhados pois não são utilizados nesta aplicação.

A CPU comunica-se com a UART através de portas de entrada e saída do mesmo modo que com o 8259. É através destas portas que a CPU recebe e envia os parâmetros referentes a UART, lendo e escrevendo nos seus registros internos. O endereço base para COM1 é 3F8h e para COM2 é 2F8h.

Os endereços das portas para acesso aos registros internos são os seguintes:

COM1: COM2:

LCR (Line Control Register)	3FBh 2FBh
DLL (Divisor Latch LSB)	3F8h 2F8h
DLM (Divisor Latch MSB)	3F9h 2F9h
MCR (Modem Control Register)	3FCh 2FCh
IER (Interrupt Enable Register)	3F9h 2F9h
LSR (Line Status Register)	3FDh 2FDh
RBR (Receive Buffer Register)	3F8h 2F8h
THR (Transmit Holding Register)	3F8h 2F8h

A repetição de alguns endereços nesta lista são explicados pelo fato de que valores diferentes escritos no LCR podem alterar a função de alguns registros.

APRENDA INFORMÁTICA SEM SAIR DE CASA

CURSO RÁPIDO, ECONÔMICO E EFICAZ

Sistema de Apostilas. Basta seguir os exercícios treinando diretamente em seu PC.

COMANDOS TRADUZIDOS PARA NOSSO IDIOMA

Opção: Disquete 5 1/4" com Exercícios

Mande seu pedido para: R. Conde de Bomfim, n.º 422/418 - Tijuca - Rio CEP 20520 A/C Ricardo Flores

Desejo receber os itens assinalados abaixo. Pagarei o valor correspondente ao total do pedido, mais as despesas de remessa, na agência de correio da minha cidade.

Item	Preço	Valor %
Lotus 123	<input type="checkbox"/> 330.000,	<input type="checkbox"/> 65.000,
Quattro Pro	<input type="checkbox"/> 330.000,	<input type="checkbox"/> 65.000,
dBase III Plus Interativo	<input type="checkbox"/> 330.000,	<input type="checkbox"/> 65.000,
dBase III Plus Programado	<input type="checkbox"/> 330.000,	<input type="checkbox"/> 65.000,
Ventura Publisher	<input type="checkbox"/> 330.000,	<input type="checkbox"/> 65.000,
WordStar 5.0/6.0	<input type="checkbox"/> 330.000,	<input type="checkbox"/> 65.000,

NOME: TEL:
 ENDEREÇO:
 CIDADE: EST: CEP:
 ASSINATURA:

Pulsar Informática

Software PC

fone (011) 856.0676

DOMÍNIO PÚBLICO e SHAREWARE

- aplicativos - utilitários
- jogos - gráficos
- processadores - textos
- for windows
- desktop - etc

SÃO MAIS DE 2500 PROGRAMAS.
 solicite grátis seu super catálogo geral

HARDWARES PARA PC

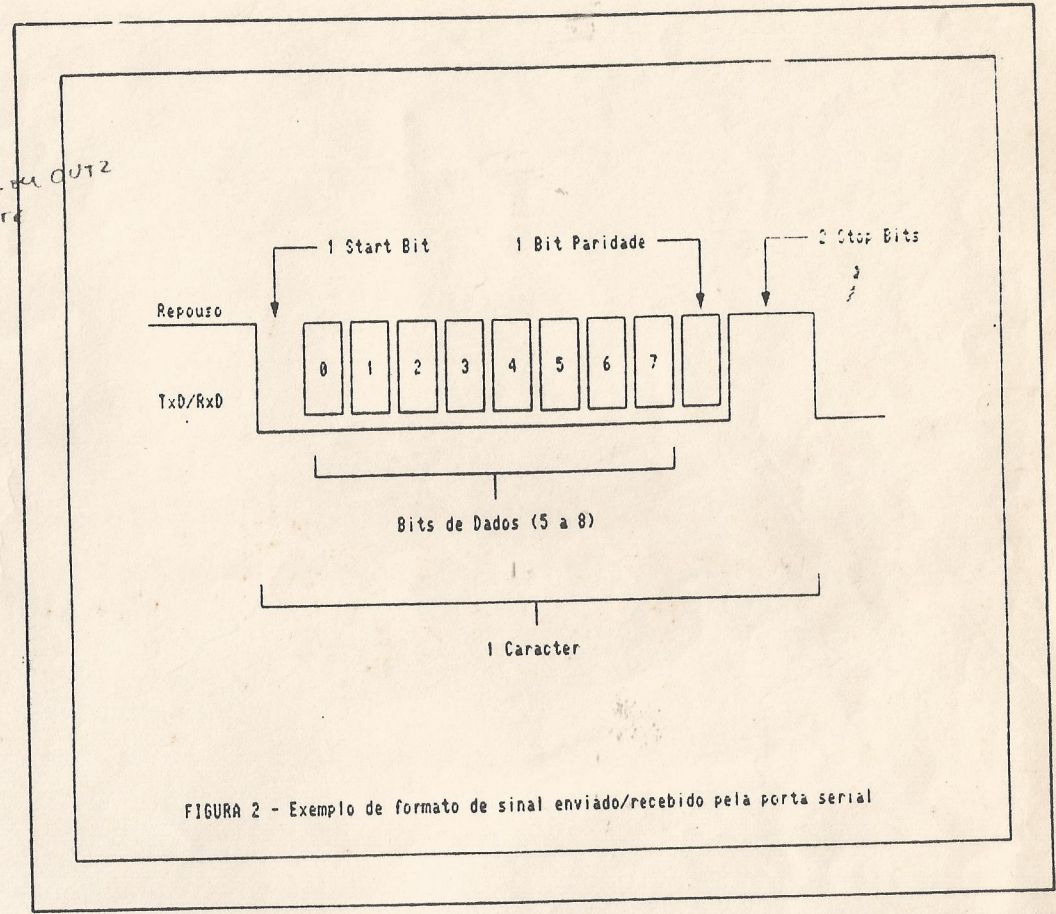
- sound blaster
- sound blaster pro
- scanner
- impressoras
- fax/modem
- mouse

PULSAR SOFT Club - clube de usuário para linha PC, solicite informações.

consulte nossos preços antes de comprar

*** se seu problema é software, nós temos uma solução *** *** o mais completo catálogo e os menores preços ***

Rua : Santa Eudóxia, 321 - SPaulo - SP - cep 02533 - 010



```

.....
Programa-Exemplo : Telefone.C
Demonstra comunicacao atraves de interrupcao
Wladimir Lopes Silva
Linguagem : Turbo C 2.0
Video : CGA,EGA,VGA,HERCULES
.....

#include <stdio.h>
#include <dos.h>
#include <conio.h>

int porta = 1; /* 1 = COM1: 2 = COM2: */
int interrupcao;

int endereco_base = 0x2F8;
int endereco_porta = 0x3F8 para COM1: 2F8 para COM2: */
RBR, /* Receive Buffer Register */
THR, /* Transmit Holding Register */
LCR, /* Line Control Register */
IER, /* Interrupt Enable Register */
MCR, /* Modem Control Register */
LSR, /* Line Status Register */
DLL, /* Divisor Latch LSB */
DLM, /* Divisor Latch MSB */

OLAB = 0x80, /* Divisor Latch Access Bit */

PARIDADE = 0x0, /* 0000 0000 no LCR bit 3 */

STOP_BIT = 0x0, /* 0000 0000 no LCR bit 2 */

TAMANHO = 0x03, /* 0000 0011 no LCR bits 0 e 1 */

DTR = 0x01, /* 0000 0001 no MCR bit 0 */

RTS = 0x02, /* 0000 0010 no MCR bit 1 */

OUT2 = 0x08, /* 0000 1000 no MCR bit 3 */

IMR = 0x21, /* Registro Mascara Interrupcoes */

ICR = 0x20, /* Registro Comando Interrupcoes */

IRQBIT;

char byte_inferior; /* divisor taxa baud byte inferior */
char byte_superior; /* divisor taxa baud byte superior */

char buffer[1023]; /* buffer circular */
int ultimo_salvo;
int ultimo_lido;
char caracter_lido;
char tecla;
int sair,x1,y1,x2,y2;

void interrupt (*velha_interrupcao);
/* ponteiro que guarda o endereco */
/* da interrupcao original da */
/* porta serial */

void interrupt nova_interrupcao()
/* nova interrupcao porta serial */

```

```

/*(ultimo_salvo = 1023)
ultimo_salvo = 0;
else
ultimo_salvo++;

buffer[ultimo_salvo] = inportb(RBR);
/* guarda no buffer o caracter */
/* recebido pela porta serial */

outportb(ICR,0x20); /* fim de interrupcao para 8259 */

(*velha_interrupcao); /* chama interrupcao original */

void setup_serial() /* configura a porta serial */
{
outportb(IER,0); /* desabilita interrupcao da UART */

velha_interrupcao = getvect(interrupcao);
/* guarda endereco da */
/* interrupcao original */
/* da porta serial */

setvect (interrupcao,nova_interrupcao);
/* seta nova interrupcao */
/* para porta serial */

outportb(LCR,(inportb(LCR) | OLAB));
/* habilita carregar taxa */
/* baud (bits/seg) */

byte_superior = 0x01; /* byte superior do divisor 384 */
byte_inferior = 0x80; /* byte inferior do divisor 384 */
/* 384 e o divisor para 300 Bauds */

outportb(DLM,byte_superior); /* seta taxa de 300 bauds */
outportb(DLL,byte_inferior);

outportb(LCR,(TAMANHO | PARIDADE | STOP_BIT));
/* seta tamanho, paridade */
/* e stop bits */

outportb(MCR,(DTR | RTS | OUT2));
/* habilita DTR,RTS e OUT2 no MCR */

outportb(IMR,(inportb(IMR) & (~ IRQBIT)));
/* Habilita interrupcao */
/* no 8259 */

outportb(IER,0x01);
/* habilita interrupcao quando */
/* dado e recebido na UART */

main() /* programa principal */
{
interrupcao = 13 - porta;
/* valor 12 = 0Ch para COM1; */
/* valor 11 = 0Bh para COM2: */

endereco_porta = endereco_base | (porta << 8);

RBR = endereco_porta; /* endereços das portas conforme */
THR = endereco_porta; /* texto */

```

Ver no HCLP { outportb inportb

manter da porta serial que chegou

2F8: 0000 0010 1111 1000
 2 << 8
 0000 0010 0000 0000
 2F8

```

LCR = endereco_porta + 3; /* 2F8 + 3 = 2FB */
IER = endereco_porta + 1; /* 2F8 */
MCR = endereco_porta + 4; /* 2FC */
LSR = endereco_porta + 5; /* 2FD */
DLL = endereco_porta; /* 2F8 */
DLM = endereco_porta + 1; /* 2F9 */

IRQBIT = 0x20 >> porta; /* porta */
/* 0000 0000 >> 0000 1000 */
setup_serial(); /* configura porta serial */ /* habilita */

clrscr(); /* limpa a tela */ /* de 16 bits x porta = 1 */

printf("\n\nPrograma Telefone - Digite ALT-X para sair\n\n");
/* monta tela */
gotoxy(1,5);
printf("Recebendo : ");
gotoxy(1,13);
printf("Transmitindo : ");
x1 = y1 = x2 = y2 = 1;
while( !sair ) /* sair sempre com 3 */
{
if(ultimo_salvo != ultimo_lido)
{
if(ultimo_lido = 1023)
ultimo_lido = 0;
else
ultimo_lido++;
caracter_lido = buffer[ultimo_lido];
window(1,7,78,12);
gotoxy(x1,y1);

putchar(caracter_lido); /* mostra caracter recebido */
x1 = wherex();
y1 = wherey();
}

if( kbhit() )
{
tecla = getch(); /* digite ALT-X para sair */
if(tecla == 0)
{
tecla = getch();
if (tecla == 45)
sair = 1;
}

window(1,15,78,20);
gotoxy(x2,y2);
putchar(tecla);
x2 = wherex();
y2 = wherey();
outportb(THR,tecla); /* transmite caracter digitado */

outportb(IER,0x0); /* finalizando programa */
outportb(IMR,(inportb(IMR) | IRQBIT));
outportb(MCR,0x0);
setvect (interrupcao,velha_interrupcao);
/* restaura interrupcao */
/* original ao sair */
}
}

```

YNDIANA

LIGUE JÁ!
CONSULTE NOSSOS PREÇOS.
TEMOS A MAIS COMPLETA
LINHA DE EQUIPAMENTOS E
PERIFÉRICOS.

- 286 20 MHZ 1 MB HD 40 • 386 SX 33 MHZ 2 MB HD 40 •
- 386 DX 40 MHZ 4 MB HD 80 • 486 DX 50 MHZ 4 MB HD 120 256
- CACHE • MONITOR CGA BRANCO • MONITOR VGA
- MONOCROMÁTICO • MONITOR VGA COLOR • MONITOR SVGA
- COLOR • PLACA VGA 256 K • PLACA SVGA 512 K • PLACA
- SVGA 1024 K • PLACA CGA DUAL • PLACA IDE COM
- CABO • PLACA IDE PLUS COM CABO • HD 40 MB IDE • HD 80
- MB IDE • HD 120 MB IDE • DRIVE 1.2 MB • DRIVE 1.44 MB FAX •
- MODEM 2400/9600 BPS • IMPRESSORA LX 810L COM
- CABO • IMPRESSORA LQ - 570 COM CABO

NA COMPRA DE UM EQUIPAMENTO
COMPLETO, GANHE
UM MOUSE E 3 JOGOS

TEMOS TAMBÉM A MAIS COMPLETA
BIBLIOTECA DE JOGOS E APLICATIVOS DE
DOMÍNIO PÚBLICO

- INDIANA JONES OF FATE
- PIT FIGHTER
- ATLANTIS
- MONKEY ISLAND II
- LEISSURE SUIT LARRY V
- WING COMMANDER II
- DRAGONS LAIR III
- THE SECRET WEAPONS OF
- LEMMINGS III
- LUFTWAFFE
- EYE OF BEHOLDER II
- THE LEGEND OF KIRANDYA

JOGOS E APLICATIVOS (DISCO INCLUSO)

- 5 1/4 DD.....CR\$ 34.000,
- 5 1/4.....CR\$ 47.000,
- 3 1/2 HD.....CR\$ 65.000,

SOLICITE CATÁLOGO GRÁTIS.

NA YNDIANA VOCÊ ENCONTRA
QUALIDADE, PONTUALIDADE E OS
MELHORES PREÇOS

YNDIANA
INFORMÁTICA LTDA.
 AV. COM. ALBERTO BONFIGLIOLI, 453 - 05593-000
 SÃO PAULO - SP. TEL/FAX: (011) 869-1112

O PROGRAMA EXEMPLO

O programa exemplo TELEFONE.C foi implementado em Turbo C, versão 2.0, mas poderá ser compilado sem maiores problemas em versões posteriores. Ele é uma pequena referência para os programadores que desejam desenvolver programas utilizando a porta serial.

Este programa envia pela porta serial um caracter digitado, além disso mostra na tela se algum caracter foi recebido pela porta serial. A porta definida no programa é a COM1, para mudar para COM2 basta modificar a variável "porta" para o valor 2.

A função "setup_serial()" é responsável pela configuração da porta serial, ou seja, define:

taxa de recepção/transmissão dos dados em bauds (bits/seg);

paridade;

tamanho do caracter recebido/transmitido;

número de stop bits.

A porta serial foi configurada com taxa de transmissão/recepção de 300 bauds, sem paridade, tamanho dos dados em 8 bits e 1 stop bit. Esta configuração pode ser alterada através da mudança dos valores das variáveis "byte_superior", "byte_inferior", "PARIDADE", "TAMANHO" e "STOP_BIT". Além disso, esta função habilita DTR, RTS, OUT2 e as interrupções.

Quando um caracter é recebido pela porta serial, é gerada uma interrupção de hardware. O tratamento original da interrupção foi modificado para atender as necessidades do sistema, isto pode ser feito através das funções "getvect()" e "setvect()". Através destas funções uma nova rotina de

tratamento da porta serial pode ser colocada no endereço da rotina original.

A nova rotina armazena o caracter recebido em um buffer circular para ser mostrado na tela posteriormente. O tamanho desta rotina deve ser mínimo, além disto não devem ser utilizadas funções que façam chamadas ao DOS, isto para evitar problemas com a não reentrância do mesmo.

Caso alguma tecla seja atuada, o caracter será transmitido pela porta serial, conforme já foi dito anteriormente. Este caracter também é mostrado na tela.

O programa pode ser finalizado digitando-se ALT-X.

A CONEXÃO ENTRE OS COMPUTADORES

Para conectar os microcomputadores será necessário a confecção de um cabo para interligá-los. O cabo poderá utilizar apenas 3 pinos da porta serial:

pino 2 - TxD - Transmissão de Dados - é através deste pino que os dados são enviados pela porta serial;

pino 3 - RxD - Recepção de Dados - é através deste pino que os dados são recebidos pela porta serial;

pino 7 - GND - Terra - é o pino usado como referência para os outros sinais.

Estas linhas de sinal seguem o padrão RS-232C. Este padrão define as características de todos os sinais no conector da porta serial do microcomputador. O modo como estes pinos devem ser interligados na confecção do cabo é mostrado na figura 3.

Caso o usuário do programa não tenha a facilidade de conectar 2 microcomputadores para testar o

programa, ele poderá simular esta condição utilizando apenas um microcomputador. Isto é possível jumpando externamente o pino 2 (TxD) com o 3 (RxD) no conector de 25 pinos da porta serial, neste caso o caracter transmitido pelo pino 2 será recebido pelo pino 3.

Deve-se ter o cuidado de não jumpar os pinos incorretamente, pois isto pode danificar a porta serial do microcomputador.

Caso exista um mouse conectado a porta serial o programa irá mostrar na tela os sinais enviados pelo mesmo.

CONSIDERAÇÕES FINAIS

Este texto procurou retratar uma implementação de comunicação entre microcomputadores padrão IBM-PC utilizando interrupção de hardware e o acesso direto às portas de programação do hardware envolvido. Este tipo de implementação permite um acesso mais rápido aos dados evitando deste modo perda de informação, além de possibilitar altas taxas de transmissão. Poderia-se implementar esta comunicação através de interrupções de software (DOS ou BIOS), mas a performance do sistema reduziria bastante.

WLADIMIR LOPES SILVA é formado em Engenharia Elétrica pela UFMG. Trabalha como engenheiro de automação e programa em Basic, Pascal, C, Fortran, Autolisp, Dbase III e Clipper

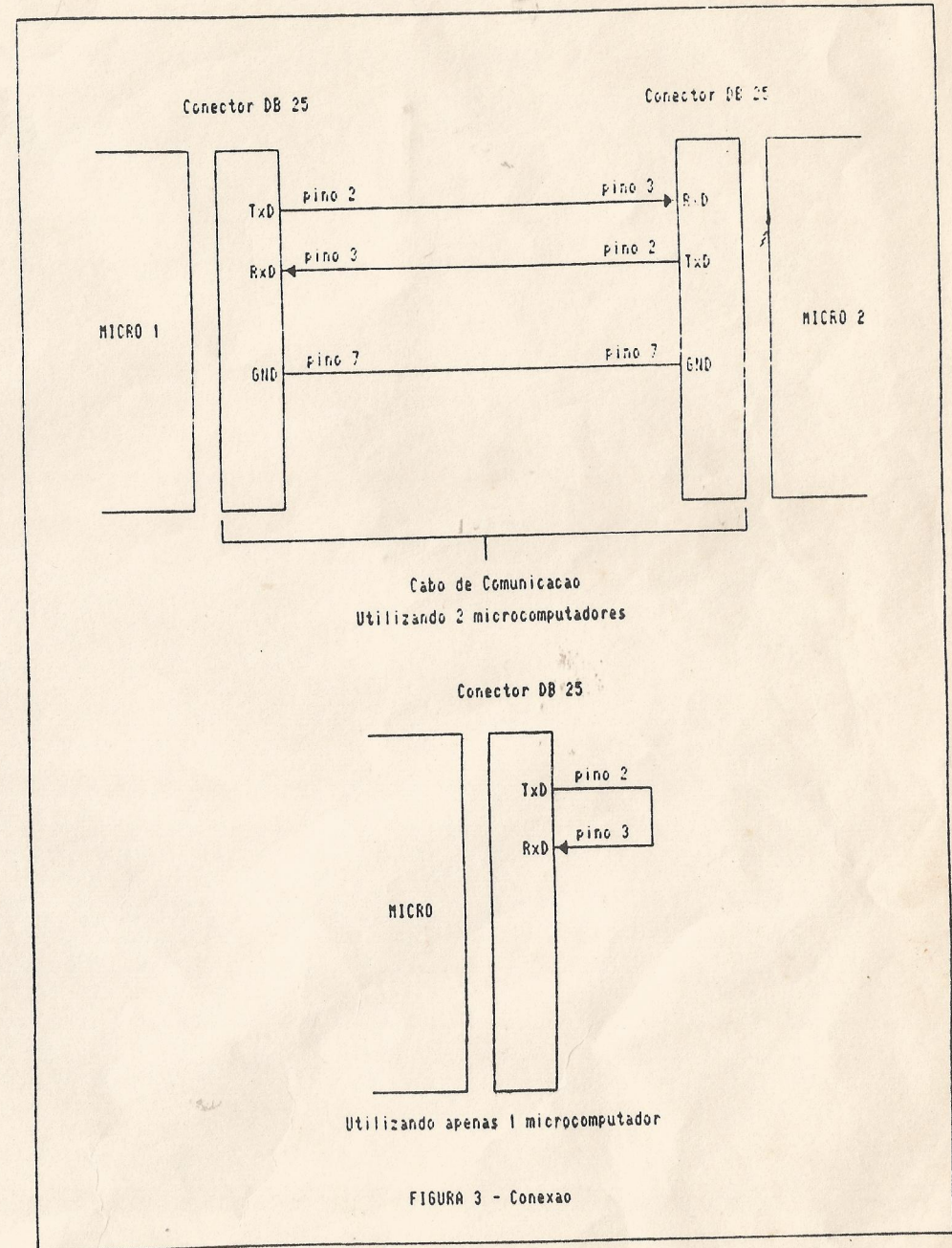


FIGURA 3 - Conexão

LOV

LIVRARIA

CIÊNCIA NOVA

• EXCLUSIVAMENTE LIVROS DE INFORMÁTICA
• LIVROS E REVISTAS IMPORTADAS

• COMPLETA LINHA DE SUPRIMENTOS:
• DISQUETES
• FORMULÁRIOS
• ETIQUETAS
• PORTA-DISQUETES
• CAPAS

• ESTABILIZADOR
• FILTRO DE LINHA
• CAIXA COMPUTADORA ETC...

• Despachamos p/ todo o Brasil
• Remessas feitas por reembolso Postal

Av. Presidente Vargas, 542/301 — Tel.: (021) 233-4045