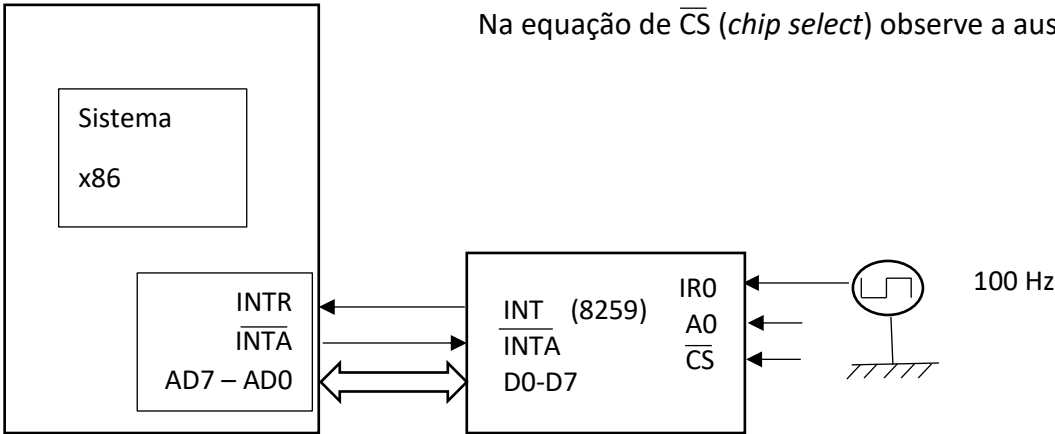


Fazer um relógio digital que conta de 00:00:00 até 23:59:59. O circuito a ser utilizado está mostrado abaixo. Suponha que o PIC8259 está no endereço de I/O = 20h (em 16 bits de endereçamento, 20h = 00000000000100000<sub>2</sub> e que IRO equivale à interrupção 18h (Suponha uP 8088).

Registrador do PIC8259	Valor do bit A0 (bus de endereços)	Conteúdo a ser armazenado no registrador
ICW1	0	0x13
ICW2	1	0x18 (enunciado)
ICW4	1	1
OCW1	1	11111110 <sub>2</sub> = 0xFE (veja IRO na figura)
OCW2	0	0x20 (não confundir com o endereço de I/O = 20h). Aqui EOI normal = 0x20

$\overline{CS} = A15+A14+A13+A12+A11+A10+A9+A8+A7+A6+\overline{A5}+A4+A3+A2+A1+\overline{IO/\overline{M}}$ , na qual ‘+’ é OU-Lógico. Na equação de  $\overline{CS}$  (chip select) observe a ausência de A0.



```
cli      ; faz IF=0. Não recebe INTR
mov ax, dados
mov ds, ax
mov ax, pilha
mov ss, ax
mov sp, topopilha
; Configura o PIC8259
mov al, 13h
out 20h, al; ICW1
mov al, 18h
out 21h, al; ICW2
mov al, 1
out 21h, al; ICW4
mov al, 0xFE
out 21h, al; OCW1
; Preenche Tabela de Interrupção
xor ax, ax
mov es, ax
mov word[es:18h*4], relógio
mov word[es:18h*4+2], cs
sti      ; if = 1. Recebe INTR
```

```
;Interrupt Service Routine (ISR)
relógio:
; suponha relógio = 0xABCD
inc byte[tique]
cmp byte[tique], 100
jb fim
mov byte[tique], 0
inc byte[segundo]
cmp byte[segundo], 60
jb fim
mov byte[segundo], 0
inc byte[minuto]
cmp byte[minuto], 60
jb fim
mov byte[minuto], 0
inc byte[hora]
cmp byte[hora], 24
jb fim
mov byte[hora], 0
fim:
mov al, 20h; EOI normal
out 20h, al;
iret
segment dados
tique      db 0;
segundo    db 0;
```

```
minuto     db 0;
hora       db 0;
segment pilha stack
resb 100h;
topopilha:
```

TABELA DE INTERRUPTÃO

INT	Endereço Físico da Memória	Conteúdo
INT 18H	0x00018x4+3	CS High
	0x00018x4+2	CS Low
	0x00018x4+1	0xAB
	0x00018x4	0xCD
INT 2	-----	-----
	-----	-----
	0x00009	IP High
	0x00008	IP Low
INT 1	0x00007	CS High
	0x00006	CS Low
	0x00005	IP High
	0x00004	IP Low
INT 0	0x00003	CS High
	0x00002	CS Low
	0x00001	IP High
	0x00000	IP Low

Programa Principal

```
volta: jmp volta
```

