Manual Técnico

Micro-ondas 8086

# Introdução

* Este manual explica de forma detalhada o funcionamento do código.
* Ao final do manual, no Anexo I pode-se encontra o código fonte comentado.
* No Anexo II há um fluxograma explicando o funcionamento básico do programa.

# Interrupções

O programa trabalha com 3 interrupções básicas. Nos endereços 90h, 91h e 92h, nomeadas respectivamente como Adiciona30, DecrementaPotencia, PausaCancelaTudo, como observado no trecho de código abaixo:

MOV AX,CS

MOV DS,AX ; Segmento do vetor (0000h) em DS

MOV DX,offset Adiciona30 ;Declaracao de interrupcao em 90h

MOV AL,90H

MOV AH,25H

INT 21H

MOV DX,offset DecrementaPotencia ;Declaracao de interrupcao em 91h

MOV AL,91H

INT 21H

MOV DX,offset PausaCancelaTudo ;Declaracao de interrupcao em 92h

MOV AL,92H

INT 21H

Vale mencionar que foi usado a INT21h/AL=25h, como forma de salvar as interrupções, como explicado em laboratório.

Os tratadores das interrupções basicamente alteram o estado do programa:

Adiciona30:

MOV AL, ESTADO\_ATUAL

MOV ESTADO\_ANTERIOR, AL

MOV ESTADO\_ATUAL,01h

IRET

DecrementaPotencia:

MOV AL, ESTADO\_ATUAL

MOV ESTADO\_ANTERIOR, AL

MOV ESTADO\_ATUAL,02h

IRET

PausaCancelaTudo:

MOV AL, ESTADO\_ATUAL

MOV ESTADO\_ANTERIOR, AL

MOV ESTADO\_ATUAL,04h

IRET

Para entender o funcionamento de cada estado basta ir para a seção Estados.

# Programa Principal

Após salvar as interrupções, o programa segue para a sua parte principal:

Aguarda\_Usuario:

CMP ESTADO\_ATUAL,01h

JE Add30s

CMP ESTADO\_ATUAL,02h

JE VariaPot

CMP CANCELAPAUSA,02h

JE Cancela\_Tudo

CMP CANCELAPAUSA,01h

JE Pausa\_Timer

CMP ESTADO\_ATUAL,04h

JE verificaPausaCancela

CMP ESTADO\_ATUAL,00h

JE Decrementador

CMP ESTADO\_ATUAL,0FFh

JE AtualizaRelogio

Esta parte do programa fica em um loop infinito, onde fica constantemente verificando os estados da máquina. O estado inicial do programa é 0FFh, que apenas atualiza a hora na variável TELALED, e retorna para Aguarda\_Usuario. Na medida que as interrupções são feitas, os estados se alteram, e programa dá o jump condicional para a função correspondente abaixo.

# Funções

O programa possui 13 funções diferentes, muitas delas não eram estritamente necessárias, mas foram criadas para facilitar a leitura do código:

1. Add30s: Trabalha de duas maneiras. A primeira se dá quando o micro-ondas estiver em stand-by, quando Int90h for ativado, 30 segundos são adicionados ao TIMER, e começam a ser decrementados, segundo por segundo na função 2. A segunda acontece quando a contagem está pausada (função 9), e Int90h é ativado, recomeçando a decrementar a partir do valor em TIMER.
2. Decrementador: Quando a esta função está atuando, LED1 encontra-se ligado. A cada segundo, a função decrementa TIMER, e joga o valor como um valor ASCII na variável TELALED.
3. VariaPot: Altera a potência do sistema. Só pode ser utilizada com o micro-ondas em stand-by.
4. RestauraPot: A função 3, pode ser usada várias vezes, como o usuário desejar. Mas quando a potência chega a um valor nulo, a função RestauraPot restaura o seu valor para 100d.
5. Cancela\_Tudo: Dá um jump para função. 10. Após dois cliques em Int92h, é usada, isto é, CANCELAPAUSA igual a 02h.
6. verificaPausaCancela: Verifica quantas vezes a Int92h foi usada. Altera o subestado CANCELAPAUSA.
7. Setapausa: Seta CANCELAPAUSA 01h
8. Setacancela: Seta CANCELAPAUSA 02h
9. Pausa\_Timer: Quando CANCELAPAUSA for igual 01h, esta função é utilizada. A mesma é um loop, infinito, que fica verificando a alteração de estados.
10. Reseta\_Tudo: Função utilizada em duas situações. Quando o valor TIMER atinge 00h após ser decrementado. E quando a interrupção Int92h é utilizada duas vezes durante uma decrementação. Joga o programa para o estado inicial, onde aguarda o usuário.
11. AtualizaRelogio: Função utilizada quando o aparelho está em stand-by. Busca a hora do sistema fazendo uso das funções 12 e 13.
12. GET\_TIME: Através de uma interrupção do sistema (Int21h/AH 02Ch), o programa pega a hora do sistema, e salva o valor convertido pela função 13, na variável TELALED
13. CONVERT: Converte números em caracteres ASCII.

# Variáveis

O programa trabalha com seis variáveis:

TIMER DB 00h ;

CANCELAPAUSA DB 00h ;

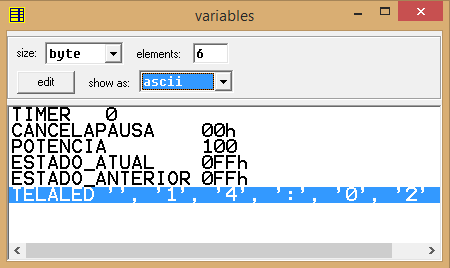
POTENCIA DB 100d ;

ESTADO\_ATUAL DB 0FFh ;

ESTADO\_ANTERIOR DB 0FFh ;

TELALED DB ' 00:00$' ; hr:min

A variável mais importante para o usuário é TELALED. A mesma indica o valor ASCII do que está acontecendo no microcontrolador. Dependendo das variáveis de estado (ESTADO\_ATUAL, ESTADO\_ANTERIOR, CANCELAPAUSA), TELALED indica algo diferente.



Quando simular o código, é importante ressaltar que as variáveis tem formatos e tamanhos diferentes, sendo que TIMER e POTENCIA devem ser setados como signed, com um elemento. As variáveis de estado são todas hexadecimais, e de um elemento apenas. TELALED é uma variável ASCII, de 6 elementos.

# Estados

O funcionamento do microcontrolador se dá pela alteração de estados da máquina, na medida que as interrupções são feitas. Com a alteração do estado da máquina, o programa voltara ao programa principal, e dará um jump condicional, para a função que cada estado corresponde. Existe uma condição especial para a interrupção em 92h. Como a mesma realiza duas funções diferentes, existe um subestado necessário para verificar se a mesma foi usada mais de uma vez seguida, como pode-se verificar abaixo:

* ESTADO\_ATUAL 00h: Decrementador
* ESTADO\_ATUAL 01h: Adiciona30s e liga
* ESTADO\_ATUAL 02h: Cancela programa atual, reseta tudo
* ESTADO\_ATUAL 04h: Verifica se o botão pause foi clicado, se não, pausa a contagem
* CANCELAPAUSA 01h: Pausa timer
* CANCELAPAUSA 02h: Reseta tudo
* ESTADO\_ATUAL 0FFh: Aguarda usuário e atualiza relógio

# Anexo I

ORG 100h

#start=DAS5332\_Temp.exe#

;#start=led\_display.exe#

;#make\_bin#

;----------------------------------------------

; Atribuicao do endereco dos tratadores de int 90h,

; int 91h,int 92h, int 93,

; na tabela de vetores de interrupcao

;

;----------------------------------------------

MOV AX,CS

MOV DS,AX ; Segmento do vetor (0000h) em ES

MOV DX,offset Adiciona30 ;Declaracao de interrupcao em 90h

MOV AL,90H

MOV AH,25H

INT 21H

MOV DX,offset DecrementaPotencia ;Declaracao de interrupcao em 91h

MOV AL,91H

INT 21H

MOV DX,offset PausaCancelaTudo ;Declaracao de interrupcao em 92h

MOV AL,92H

INT 21H

;MOV DX,offset PortaAberta ;Declaracao de interrupcao em 93h

;MOV AL,93H

;INT 21H

;----------------------------------------------

; Programa principal

;-----------------------------------------------

Aguarda\_Usuario:

CMP ESTADO\_ATUAL,01h

JE Add30s

CMP ESTADO\_ATUAL,02h

JE VariaPot

CMP CANCELAPAUSA,02h

JE Cancela\_Tudo

CMP CANCELAPAUSA,01h

JE Pausa\_Timer

CMP ESTADO\_ATUAL,04h

JE verificaPausaCancela

CMP ESTADO\_ATUAL,00h

JE Decrementador

CMP ESTADO\_ATUAL,0FFh

JE AtualizaRelogio

JMP Aguarda\_Usuario

;-----------------------------------------------

; Tratadores

;-----------------------------------------------

Adiciona30:

MOV AL, ESTADO\_ATUAL

MOV ESTADO\_ANTERIOR, AL

MOV ESTADO\_ATUAL,01h

IRET

DecrementaPotencia:

MOV AL, ESTADO\_ATUAL

MOV ESTADO\_ANTERIOR, AL

MOV ESTADO\_ATUAL,02h

IRET

PausaCancelaTudo:

MOV AL, ESTADO\_ATUAL

MOV ESTADO\_ANTERIOR, AL

MOV ESTADO\_ATUAL,04h

IRET

;PortaAberta:

;MOV AL, ESTADO\_ATUAL

;MOV ESTADO\_ANTERIOR, AL

;MOV ESTADO\_ATUAL,04h

;IRET

;-----------------------------------------------

; Funcoes

;-----------------------------------------------

Add30s:

CMP CANCELAPAUSA, 01h ;ve se ta pausado

MOV CANCELAPAUSA, 00h

MOV ESTADO\_ATUAL,00h

JE Decrementador ;se sim, despausa

MOV AL,TIMER ;pega o valor do timer

ADD AL,1Eh

MOV TIMER,AL ;adiciona 30s ao timer

Decrementador:

LEA BX, TELALED ;offset em telaled

MOV AL, 00

CALL CONVERT

MOV [BX], 01Fh ;joga um espaco vazio no primeiro digito

MOV [BX+1], AX

MOV AL, TIMER

CALL CONVERT ;converte valor do time em ascii

MOV [BX+4], AX ;joga valor em TELALED[4]

MOV AL,01H

OUT 21h,AL ;Liga LED1

CMP ESTADO\_ATUAL,00h ;SE HOUVER ALGUMA ALTERACAO PARA ADD30S

JNE Aguarda\_Usuario

MOV CX, 0FH

MOV DX, 4240H

MOV AH,86h

INT 15h ;Faz delay de 1s, usando os valores de CX DX

MOV AL,TIMER

CMP AL,00h ;verifica se contagem acabou

JE Reseta\_Tudo

ADD AL,-01h ;decrementa valor al

MOV TIMER,AL

JMP Decrementador

VariaPot:

CMP ESTADO\_ANTERIOR, 0FFh ;so pode alterar pot em stand-by

MOV AL, ESTADO\_ANTERIOR ;

MOV ESTADO\_ATUAL, AL ;

JNE Aguarda\_Usuario ;

LEA BX, TELALED ;indica onde comeca TELALED

MOV [BX], 01010000b ;Em TELALED[0]=’P’

MOV [BX+1], 01001111b ;Em TELALED[1]=’O’

MOV [BX+2], 01010100b ;Em TELALED[2]=’T’

MOV AL, POTENCIA ;

CALL CONVERT ;

MOV [BX+4], AX ;Em TELALED, escreve valor da potencia

ADD POTENCIA, -10d ;

CMP POTENCIA, 00d ;

JE RestauraPot ;se pot=0, restaura potencia=100

MOV ESTADO\_ATUAL,0FFh ;

JMP Aguarda\_Usuario ;retorna pra programa inicial

RestauraPot:

MOV POTENCIA, 100d ;restaura potencia

MOV ESTADO\_ATUAL,0FFh ;restara estado

JMP Aguarda\_Usuario

Cancela\_Tudo:

JMP Reseta\_Tudo ;reseta tudo

verificaPausaCancela:

CMP ESTADO\_ANTERIOR,0FFH ;pra não dar pause em timer=0

JE Reseta\_Tudo

CMP CANCELAPAUSA , 00h ;Verifica se int92h clicado + de 1vz

JE setapausa ;se não

JNE setacancela ;se sim

setapausa:

MOV CANCELAPAUSA, 01H

MOV ESTADO\_ATUAL,05H

JMP Aguarda\_Usuario

setacancela:

MOV CANCELAPAUSA, 02H

JMP Aguarda\_Usuario

Pausa\_Timer:

MOV AL,00H

OUT 21h,AL ; Desliga LED1

CMP ESTADO\_ATUAL,04H

JE verificaPausaCancela ; ve se int92h clicada mais de uma vez

CMP ESTADO\_ATUAL,01H

JE Aguarda\_Usuario ; volta pra rotina principal

CMP CANCELAPAUSA,01H

JE Pausa\_Timer ; Joga pra pausa\_timer

JMP Reseta\_Tudo ; Reseta tudo. CANCELAPAUSA = 02h

Reseta\_Tudo:

MOV POTENCIA, 100d ; Reseta valor da potencia

MOV AL,00H ; AL=0

OUT 21h,AL ; Desliga LED1

MOV TIMER,00h ; zera timer

MOV ESTADO\_ATUAL,0FFh ; vai pra estado de loop infinito

MOV ESTADO\_ANTERIOR,0FFh

MOV CANCELAPAUSA,00H

JMP Aguarda\_Usuario ; volta pra rotina principal

AtualizaRelogio:

LEA BX, TELALED ; BX=offset address de TELALED

CALL GET\_TIME ; chama GET\_TIME

MOV AL,TELALED ; joga valor de TELALED em AL

MOV [BX], 01Fh

JMP Aguarda\_Usuario ; volta pra rotina principal

GET\_TIME PROC

; input : BX=offset address of the string TELALED

; output : BX=current time

PUSH AX ; PUSH AX onto the STACK

PUSH CX ; PUSH CX onto the STACK

MOV AH, 2CH ; get the current system time

INT 21H

MOV AL, CH ; set AL=CH , CH=hours

CALL CONVERT ; call the procedure CONVERT

MOV [BX+1], AX ; set [BX]=hr , [BX] is pointing to hr

; in the string TELALED

MOV AL, CL ; set AL=CL , CL=minutes

CALL CONVERT ; call the procedure CONVERT

MOV [BX+4], AX ; set [BX+3]=min , [BX] is pointing to min

; in the string TELALED

POP CX ; POP a value from STACK into CX

POP AX ; POP a value from STACK into AX

RET ; return control to the calling procedure

GET\_TIME ENDP ; end of procedure GET\_TIME

CONVERT PROC

; input : AL=binary code

; output : AX=ASCII code

PUSH DX ; PUSH DX onto the STACK

MOV AH, 0 ; set AH=0

MOV DL, 10 ; set DL=10

DIV DL ; set AX=AX/DL

OR AX, 3030H ; convert the binary code in AX into ASCII

POP DX ; POP a value from STACK into DX

RET ; return control to the calling procedure

CONVERT ENDP ; end of procedure CONVERT

TIMER DB 00h

CANCELAPAUSA DB 00h

POTENCIA DB 100d

ESTADO\_ATUAL DB 0FFh

ESTADO\_ANTERIOR DB 0FFh

TELALED DB ' 00:00$' ; time format hr:min

# Anexo II

