

```
' =====  
' {$PBASIC 2.5}  
' {$STAMP BS2px,fonteAT1,fonteAT2,fonteAT3,fonteAT4,fonteAT5}  
'  
' Fonte de Alta tensão - 0-32kV D.C. com controle digital  
' Projeto de Luiz A. Feijó Jr. em 20/08/2008  
' =====
```

```
a VAR Word  
b VAR Nib  
c VAR Nib  
d VAR Nib  
e VAR Word  
f VAR Word  
g VAR Word  
h VAR Word  
pausa VAR Byte ' Define o tempo de pausa para escrita no LCD  
AD1 VAR Word ' Contém o valor do A/D 1 (8 bits)  
AD2 VAR Word ' Contém o valor do A/D 2 (8 bits)  
string VAR Word ' usado na rotina do LCD
```

```
dados CON 0 'pino de dados da CPU  
clock CON 1 'pino de clock da CPU
```

```
ADC08311 CON 8  
ADC08312 CON 9  
DAC7611 CON 5
```

```
*****  
* Inicializacao do computador *  
* (power On) *  
*****
```

```
DIRS=%0000111100111100  
OUTS=%0000100000100000
```

```
LOW 11  
GOSUB inidisp  
string=inicializando  
pausa=100  
GOSUB escreve  
HIGH 11  
a=000  
GOSUB escreve_led  
g=0
```

```

GOSUB dacdata
pisca:
FOR b=1 TO 5
PAUSE 100
LOW 11
PAUSE 200
HIGH 11
NEXT
a=300
GOSUB dacdata
GOSUB limpa
pausa=0

teste:
GOSUB canto

IF IN15=0 AND IN14=0 THEN GOSUB zera

IF IN13=0 AND IN12=0 THEN reseta

IF IN14=0 AND f<3500 THEN f=f+10

IF IN12=0 AND f<4000 THEN f=f+1000

protege1:
IF IN12=0 THEN GOTO protege1

IF f=0 THEN GOTO pula

IF IN15=0 THEN f=f-10

IF IN13=0 AND f>999 THEN f=f-1000

protege2:
IF IN13=0 THEN GOTO protege2
IF f>3500 THEN f=3500

pula:

GOSUB adpdata
IF ad1*10 > f THEN GOSUB diminui
IF ad1*10 < f THEN GOSUB aumenta

'DEBUG CLS
'DEBUG DEC ad1*10,CR
'DEBUG "valor de f: ", DEC f,CR

```

```
'DEBUG "valor lido no ADC: ",DEC ad1*10,CR  
'DEBUG "valor de g: ",DEC g,CR  
GOSUB dacdata
```

```
a=f/10  
GOSUB ESCREVE_LED  
GOSUB canto  
string=inputAD1  
GOSUB escreve  
a=AD1 DIG 2  
GOSUB ajuste  
a=AD1 DIG 1  
GOSUB ajuste  
a="."  
GOSUB dadodisp  
a=AD1 DIG 0  
GOSUB ajuste  
a="k"  
GOSUB dadodisp  
a="V"  
GOSUB dadodisp  
GOTO teste
```

```
diminui:  
IF g<10 THEN termina_diminui  
g=g-10  
termina_diminui:  
RETURN
```

```
umenta:  
g=g+10  
LOW 11  
PAUSE 30  
HIGH 11  
RETURN
```

```
zera:  
f=0  
g=f  
a=g  
GOSUB dacdata  
RETURN
```

```
reseta:  
RUN 0
```

```
*****  
* SUB rotina de conversao *  
*****
```

```
adcddata:  
LOW ADC08311  
SHIFTIN dados,clock,2,[AD1\9]  
AD1=(AD1)*2  
HIGH ADC08311  
LOW ADC08312  
SHIFTIN dados,clock,2,[AD2\9]  
AD2=(AD2)*2  
HIGH ADC08312  
RETURN
```

```
*****  
* SUB rotina do DAC *  
*****
```

```
dacdata:  
HIGH 7  
PAUSE 2  
LOW 5  
SHIFTOUT dados,clock,1,[g\12]  
LOW 7  
PAUSE 3  
HIGH 7  
HIGH 5  
RETURN
```

```
*****  
* Sub-rotinas do display LED *  
*****
```

```
' COLOCAR EM a O VALOR A SER INDICADO!  
escreve_led:  
b=a DIG 0  
c=a DIG 1  
d=a DIG 2  
e.LOWBYTE.LOWNIB=d  
e.LOWBYTE.HIGHNIB=c  
e.HIGHBYTE.LOWNIB=b  
SHIFTOUT dados,clock,1,[e\16]
```

```
PULSOUT 10,5  
RETURN
```

```
*****  
*   Sub-rotinas do display LCD   *  
*****
```

```
insdisp: 'envia instrucao ao display  
SHIFTOUT dados,clock,1,[a]  
PULSOUT 2,5  
OUT4=0  
PULSOUT 3,5  
PAUSE 5  
RETURN
```

```
dadodisp: 'envia dado ao display  
SHIFTOUT dados,clock,1,[a]  
PULSOUT 2,5  
OUT4=1  
PULSOUT 3,5  
PAUSE 5  
RETURN
```

```
limpa: 'limpa o display e volta ao home  
a=$1  
GOSUB insdisp  
PAUSE 10  
RETURN
```

```
canto:  
a=$80  
GOSUB insdisp  
PAUSE 5  
RETURN
```

```
segunda_linha:                'pula cursor para segunda linha  
a=$c0  
GOSUB insdisp  
RETURN
```

```
terceira_linha:              'pula cursor para terceira linha  
a=$190  
GOSUB insdisp  
RETURN
```

```
quarta_linha:          'pula cursor para quarta linha
a=$d0
GOSUB insdisp
RETURN
```

```
escreve: 'escreve o conteudo de string no display
READ string,a
IF a=58 THEN close
GOSUB dadodisp
IF a <> 58 THEN continua
close:
RETURN
```

```
continua: 'sub que faz parte de escreve
string=string+1
PAUSE pausa
GOTO escreve
```

```
ajuste: 'sub que corrige o valor numerico para display
a=a+48
GOSUB dadodisp
RETURN
```

```
inidisp: 'inicializacao do display
PAUSE 15
a=$38
GOSUB insdisp
a=$6
GOSUB insdisp
a=$C
GOSUB insdisp
a=$1
GOSUB insdisp
RETURN
```

```
*****
'*      Mensagens do display      *
*****
```

```
'abaixo estao as mensagens que aparecem no display
'deve-se terminar sempre com o ponto decimal.
'para separar uma da outra
```

```
inicializando DATA "Fonte AT digital:"
inputAD1 DATA      "V saida= :"
```