

**SOFTWARE DI GESTIONE DEL SISTEMA  
DI PUNTAMENTO ELETTRICO DEL RAMO  
N/S PER IBM-286**

**- PROGRAMMA PUNTA10 -**

**Montebugnoli S. - Bortolotti C.  
Cattani A. - Grueff G.**

**Rapporto interno IRA - 144/91**

**Word File: PUNTA.DOC**

## **INDICE**

PREMESSA	pagina	3
INTRODUZIONE	4	
PROGRAMMA PUNTALO	5	
BIBLIOGRAFIA	26	
NOTE	27	

## **PREMESSA**

Il presente rapporto interno riporta la descrizione del software relativo alla gestione completa del puntamento elettrico del ramo NS del radiotelescopio "CROCE del NORD", scritto per il nuovo calcolatore (IBM 286) di gestione del sistema, sostituente il precedente Apple 2 Plus. Sono state apportate, e sono tuttora in via di attuazione, modifiche ed aggiunte al sistema elettromeccanico inerente il movimento dei vasi e controllo pneumatico dei livelli del kerosene all'interno dei vasi stessi, che verranno trattate in un futuro rapporto interno.

Si ringrazia l'Ing. Orfei A. per la revisione ed i consigli dati durante la stesura di questo rapporto interno.

## **INTRODUZIONE**

Il programma PUNTAL0 gestisce il puntamento elettrico del ramo NS eseguendo automaticamente, una volta data la declinazione, le seguenti operazioni:

- 1) Posizionamento degli otto vasi di kerosene ai livelli calcolati in funzione della declinazione di puntamento, per la messa in fase delle 8 antenne NS costituenti un canale.
- 2) Introduzione degli step di cavo (ritardi variabili) per la equalizzazione dei ritardi tra i canali EW e canali NS.
- 3) Gestione dei nuovi sfasatori variabili a 5 bit, inseriti nelle catene di Media-Frequenza, per la formazione on-line dei fasci NS, correggendo anche l'offset di fase EW-NS.

NOTA: se si punta il ramo NS per osservare tramite gli interferometri sciolti (EW\*NS), sara` compito del programma TOS (Tape Operative System) di resettare a fase zero tali sfasatori, prima dell'inizio della osservazione.

## **PROGRAMMA PUNTA10**

L'esecuzione del programma PUNTA10 di gestione del puntamento elettrico del ramo NS, parte con la richiesta della declinazione (gradi, primi) in funzione della quale vengono calcolati e inseriti gli step di cavo necessari per correggere le differenze di tempo di arrivo dei due canali al correlatore, dovute alle differenze di percorso EW-NS (Sommando ai risultati anche gli offset di ritardo fissi). Il programma calcola poi le fasi da impostare nelle otto antenne che compongono ogni canale NS, e i relativi livelli dei vasi contenenti il kerosene, e provvede al loro posizionamento. Una volta posizionati i vasi (questa operazione puo` richiedere un massimo di 3 minuti per essere completata) calcola le fasi da inserire negli sfasatori variabili a 5 bit, inseriti nelle catene al livello della Media-Frequenza, per la formazione dei fasci NS correggendo tali fasi con un opportuno offset che rappresenta il termine di fase EW-NS fisso (il termine variabile con la delta viene compensato successivamente via software).

Si e` prevista la possibilita` di potere posizionare i vasi in modo manuale, sia inserendo un livello uguale per tutti, sia inserendo otto livelli diversi scelti secondo le necessita`; queste due opzioni risultano utili in fase di manutenzione o riparazione del sistema elettromeccanico di movimento.

Questo programma, e` stato scritto in Fortran ed e` linkato alle subroutines:

- 1) FORT488
- 2) FASENS
- 3) OUTSFNS
- 4) OUTPORT

Nelle pagine seguenti vengono riportate brevemente le descrizioni delle subroutines e del relativo listato, compreso quello dello stesso programma PUNTA10.

## **1) FORT488**

Del pacchetto software FORT488 della Capital Equipment Corp., vengono utilizzate in questo contesto, solo le routines di gestione di input/output (non si riporta in questa sede il relativo listato, vedi rapporto interno IRA - 143/91.

**2) FASENS, 3) OUTSFNS, 4) OUTPORT**

Questi moduli hanno il compito di calcolare le fasi da introdurre nei vari canali, per formare i fasci NS e successivamente di gestirne l'inserzione. Seguono i listati.

La routine FASENS richiede in ingresso i seguenti parametri, dati come input nel PUNTAL0:

dg = gradi (real)  
pr = primi (real)

La routine OUTSFNS richiede in ingresso i seguenti parametri:

roma\_a = indirizzo dello sfasatore (integer\*2)  
roma\_d = dato (codice a 5 bit) (integer\*2)

\$DECLARE

```
C =====
C PROGRAMMA DI PUNTAMENTO
C DEL RAMO N/S
C ANTENNA, RITADI, KEROSENE
C =====
C Nome:PUNTA10

C S. Montebugnoli 12/11/1990

C ---Questo programma e` linkato con le routine
C ---- FORT488 --> routine (scheda IEEE488) IN/OUT
C ---- FASENS| --> rifasamento fascio N/S
C ---- OUTSFNS --> gestione sfasatori a cavo NS
C ---- OUTPORT --> gestione porte I/O

integer*2 pi(8),zi(8),ab,ad,abl,jdg,jpr,i
integer*2 a,b,c,d,e,f,g,h,smot,amot,mux,sconv,flag,val
integer*2 a1,a2,a3,a4,a5,a6,a7,a8,ind,let1,let2
integer*2 a11,a12,a13,a14,a15,a16,a17,a18,z1,z2
integer*2 in,fre,dur,input,j,jp
integer*2 adr1,adr2,adr4,s1,s2,s3,s4,n1,n2,n3,n4
integer*2 irow,icolumn,ivpage,ierror,key
integer*1 zs(8)
integer*4 iseg
character*80 out
character*1 chr(80),as$,cr
character*11 punt$
equivalence(out,chr(1))
real ze,dc,dg,pr,k,ksign,b1,b2,b3,b4,b5,b6,b7,b8,ai,bi,ri
real d1(8),df(8),uno,l(8),z(8),m(8),liv,del
character*10 a$
data adr1/768/,adr2/769/,adr4/771/
data smot/3/,amot/4/,mux/5/,sconv/6/,flag/13/,let1/1/,let2/0/
data s4/8/,s3/2/,s2/1/,s1/0/
data n1/9/,n2/10/,n3/11/,n4/12/
data uno/1./
=====
C Reset di tutto il Sistema
C =====.
val=0
call output (adr4,val)
*****
ze=zenith in gradi,decimi
*****
ze=44.57
-----
smot=senso motori
amot=abilitazione motori
mux,sconv,flag,let=gestione mux/convertitore A/D
----blocco del sistema----

abl=0
call outport(abl,amot)
```

```
c =====
c GESTIONE RITARDI VARIABILI
c =====
10 call cls
11 write (*,15) 'PROGRAMMA DI PUNTAMENTO OLI (versione Fortran)'
12 write (*,15) '=====
c ---- stampa in inverso della 11 e 12 ----
iseg=45056
j=112
do 13 i=1,319,2
13 call poke (iseg,i,j)
c -----
write (*,*) '                               Puntamento automatico??? (s/n)'
read (*,'(a10)') a$
if (a$.eq.'n') go to 1000
punt$='AUTOMATICO'
write (*,15) 'Inizio procedura di puntamento elettrico N/S'
write (*,*) '
write (*,16) ' -----'
write (*,16) ' Fare la prova (B),(E) nei canali N/S'
write (*,16) ' -----'
fre=1500
dur=10
call beep2 (fre,dur,in)
write (*,17) ' Fatto ??? (s/n)'
15 format (11x,a46)
16 format (11x,a40)
17 format (25x,a16)
read (*,'(a1)')a$
if (a$.eq.'n') go to 10
18 write (*,*) 'DELTA (gradi,primi) ?'
read (*,*) dg,pr
if (dg.gt.90.or.dg.lt.0) go to 10
if (pr.ge.60) go to 10
jdg=int(dg)
jpr=int(pr)
c -- scrittura su disco della delta 'puntata' -----
open(unit=10,file='\croce\punts.dat',access='direct',recl=50,
&form='formatted')
write(10,19,rec=1) jdg
write(10,19,rec=2) jpr
19 format(2x,i2)
c -----
dc=dg+(pr/60)
bi=dc/57.29578
ri=ze/57.29578
ai=(bi-ri)
c ---- inizio procedura "ritardi" -----
c **** 4 SUD ***
k=1
b1=8*(k-4.5)*sin(ai)
a1=nint(b1)
a1=a1+23
```

```
if(a1.ge.32) go to 20
fre=800
dur=4
call beep2(fre,dur,in)
call outport (a1,s4)
go to 30
val=a1+16
call outport (val,s4)
-----
c
c **** 3 SUD ****
30 k=2
b2=8*(k-4.5)*sin(ai)
a2=nint(b2)
a2=a2+23
fre=1000
dur=4
call beep2(fre,dur,in)
if (a2.ge.32) go to 40
call outport (a2,s3)
go to 45
40 val=a2+16
call outport(val,s3)
-----
c
c **** 2 SUD ****
45 k=3
b3=8*(k-4.5)*sin(ai)
a3=nint(b3)
a3=a3+15
fre=1200
dur=4
call beep2(fre,dur,in)
call outport (a3,s2)
-----
c
c **** 1 SUD ****
k=4
b4=8*(k-4.5)*sin(ai)
a4=nint(b4)
a4=a4+7
fre=1400
dur=4
call beep2(fre,dur,in)
call outport (a4,s1)
-----
c
c **** 1 NORD ****
k=5
b5=8*(k-4.5)*sin(ai)
a5=nint(b5)
a5=a5+7
fre=1600
dur=4
call beep2(fre,dur,in)
call outport (a5,n1)
-----
c
c **** 2 NORD ****
k=6
```

```
b6=8*(k-4.5)*sin(ai)
a6=nint(b6)
a6=a6+15
fre=1800
dur=4
call beep2(fre,dur,in)
call outport (a6,n2)
-----
c
c      **** 3 NORD ****
k=7
b7=8*(k-4.5)*sin(ai)
a7=nint(b7)
a7=a7+23
fre=2000
dur=4
call beep2(fre,dur,in)
if (a7.ge.32) go to 60
call outport (a7,n3)
go to 70
60
val=a7+16
call outport(val,n3)
-----
c
c      **** 4 NORD ****
70
k=8
b8=8*(k-4.5)*sin(ai)
a8=nint(b8)
a8=a8+23
fre=2200
dur=4
call beep2(fre,dur,in)
if (a8.ge.32) go to 80
call outport (a8,n4)
go to 90
80
val=a8+16
call outport (val,n4)
-----
c
90
call beep2 (fre,dur,in)
call cls
write(*,95)jdg,jpr
95
format ('RIT. PUNTATI PER DELTA=',i2,',',i2,'(gradi,primi)')
write (*,*)'IN UNITA` DI LUNGHEZZE D'ONDA'
write (*,*)' 4 SUD---',a1
write (*,*)' 3 SUD---',a2
write (*,*)' 2 SUD---',a3
write (*,*)' 1 SUD---',a4
write (*,*)' 1 NORD--',a5
write (*,*)' 2 NORD--',a6
write (*,*)' 3 NORD--',a7
write (*,*)' 4 NORD--',a8
write (*,*)'CONTROLLARE E CORREGGERE,FASE E AMPIEZZA'
write (*,*)'(PROVA "B" ED "E"), NEI CANALI N/S'
100
write (*,*)'Fatto???? (s/n)'
read (*,(a10)) a$
if (a$.eq.'n'.or.a$.eq.'N') go to 100
----- calcolo dei livelli dei vasi -----
```

```
do 320 i=1,8
d1(i)=(i-4.5)*10*sin((ze-dc)*3.141592/180)
d1(i)=d1(i)/299.793*408
val= int(d1(i))
ksign=sign(uno,d1(i))
df(i)=ksign*(abs(d1(i))-abs(val))
df(i)=df(i)*71.1+15
280 if(df(i).le.14.99) go to 292
290 if(df(i).gt.86.1) go to 294
go to 310
292 df(i)=df(i)+71.1
go to 280
294 df(i)=df(i)-71.1
go to 290
c ---- calcolo della posizione con 10 bit ---
310 df(i)= df(i)*10.2
pi(i)=nint(df(i))
c write (*,*) pi(i)
320 continue
c =====
c GESTIONE MOVIMENTO VASI
c =====
c ---- inizio gestione automatica ----
370 continue
data a/1/,b/1/,c/1/,d/1/,e/1/,f/1/,g/1/,h/1/
data a11/1/,a12/2/,a13/3/,a14/4/,a15/5/,a16/6/,a17/7/,a18/8/
data ab/255/,ad/255/,ind/1/
368 call cls
write (*,*) '
iseg=45056
do 369 i=1,160,2
j=15
369 call poke (iseg,i,j)
call beep2(fre,dur,in)
write (*,371)a11,a12,a13,a14,a15,a16,a17,a18
371 format (' CANALE:',15x,i1,6x,i1,6x,i1,6x,i1,6x,i1,6x
: ,i1,6x,i1,6x)
write (*,372)jdg,jpr
372 format (' Delta=',i2,':',i2,11x,'|',6x,'|',6x,'|',6x,'|'
: ,6x,'|',6x,'|',6x,'|')
c ---- stampa su video della 371---
iseg= 45056
j=112
do 373 i=321,480,2
373 call poke (iseg,i,j)
do 374 i=1,8
l(i)= float(pi(i))/10.2
374 continue
write (*,375)l(1),l(2),l(3),l(4),l(5),l(6),l(7),l(8)
375 format(' Posizione comandata',2x,f4.1,3x,f4.1,3x,f4.1
: ,3x,f4.1,3x,f4.1,3x,f4.1)
c -----
iseg=45056
j=112
do 376 i=641,799,2
```

```
376    call poke (iseg,i,j)
c      ---- stampa highlight degli * ----
      iseg=45056
      j=15
      do 377 i=801,959,2
377    call poke (iseg,i,j)
      irow=6
      icolm=1
      ivpage=0
      call putcur(irow,icolm,ivpage,ierror)
c      ---- CONTROLLO MUX-A/D-MOVIMENTO ====
      fre=1000
      dur=8
      as$='*'
c      ---- Gestione Canale 1 ----
380    continue
      val=0
      z1=0
      z2=0

c      -----
      val=0
      call outport (val,mux)
c      ---- start conversion ----

      val=1
      call outport (val,sconv)

c      ---- lettura flag ----
400    call output (adr2,flag)
      val=input (adr1)
      if (val.eq.1) go to 400
c      ---- lettura pos.vaso ----
      call output (adr2,let1)
      z1=input (adr1)
      z1=iand(z1,3)
c      z1=ishft(z1,-2)
      call output (adr2,let2)
      z2=input (adr1)
      zi(1)=z1+(z2*4)
c      write (*,*) pi(1),zi(1)
      if (a.eq.0) go to 430
      if((pi(1)-zi(1)).eq.0) go to 410
      if((zi(1)-pi(1)).lt.0) go to 420
      go to 430
410    ab=ab-1
      call outport (ab,amot)
      call beep2(fre,dur,in)
c      ---- stampa posiz. raggiunta ----
      iseg=45056
      i=844
      jp=ichar(as$)
      call poke (iseg,i,jp)
      a=0
```

```
        go to 430
420    ad=ad-1
430    continue
c      ---- Gestione Canale 2 ----

        val=1
        call outport (val,mux)

c      ---- start conversion ----

        val=1
        call outport (val,sconv)

c      ---- lettura flag ----
432    call output (adr2,flag)
        val=input (adr1)
        if (val.eq.1) go to 432
c      ---- lettura posizione vaso ----
        call output (adr2,let1)
        z1=input (adr1)
        z1=iand(z1,3)
c      z1=ishft(z1,-2)
c      write (*,*)z1
        call output (adr2,let2)
        z2=input (adr1)
        zi(2)=z1+(z2*4)
        if (b.eq.0) go to 480
        if ((pi(2)-zi(2)).eq.0) go to 440
        if ((zi(2)-pi(2)).lt.0) go to 450
        go to 480
440    ab=ab-2
        call outport (ab,amot)
        call beep2(fre,dur,in)
c      ---- stampa su schermo pos. raggiunta ----
        iseg=45056
        i=858
        jp=ichar(as$)
        call poke (iseg,i,jp)
        b=0
        go to 480
450    ad=ad-2
480    continue
c      ---- Gestione Canale 3 ----

        val=2
        call outport (val,mux)

c      ---- start conversion ----

        val=1
        call outport (val,sconv)
```

```
c      ---- lettura flag ----
482    call output (adr2,flag)
      val=input (adr1)
      if (val.eq.1) go to 482
c      ---- lettura posizione vaso ----
      call output (adr2,let1)
      z1=input (adr1)
      z1=iand(z1,3)
c      z1=ishft(z1,-2)
      call output (adr2,let2)
      z2=input (adr1)
      zi(3)=z1+(z2*4)
      if (c.eq.0) go to 530
      if ((pi(3)-zi(3)).eq.0) go to 500
      if ((zi(3)-pi(3)).lt.0) go to 510
      go to 530
500    ab=ab-4
      call outport (ab,amot)
      call beep2(fre,dur,in)
c      ---- stampa su schermo pos. raggiunta ----
      iseg=45056
      i=872
      jp=ichar(as$)
      call poke (iseg,i,jp)
      c=0
      go to 530
510    ad=ad-4
530    continue
c      ---- Gestione Canale 4 ----
      val=3
      call outport (val,mux)

c      ---- start conversion ----
      val=1
      call outport (val,sconv)

c      ---- lettura flag ----
532    call output (adr2,flag)
      val=input (adr1)
      if (val.eq.1) go to 532
c      ---- lettura posizione vaso ----
      call output (adr2,let1)
      z1=input (adr1)
      z1=iand (z1,3)
c      z1=ishft(z1,-2)
      call output (adr2,let2)
      z2=input (adr1)
      zi(4)=z1+(z2*4)
      if (d.eq.0) go to 580
      if ((pi(4)-zi(4)).eq.0) go to 550
```

```
      if ((zi(4)-pi(4)).lt.0) go to 560
550  go to 580
      ab=ab-8
      call outport (ab,amot)
      call beep2(fre,dur,in)
c      ---- stampa su schermo pos. raggiunta----
      iseg=45056
      i=886
      jp=ichar(as$)
      call poke (iseg,i,jp)
      d=0
      go to 580
560  ad=ad-8
580  continue
c      ---- Gestione Canale 5 ----

      val=4
      call outport (val,mux)

c      ---- start conversion ----

      val=1
      call outport (val,sconv)

c      ---- lettura flag ----
583  call output (adr2,flag)
      val=input (adr1)
      if (val.eq.1) go to 583
c      ---- lettura posizione vaso ----
      call output (adr2,let1)
      z1=input (adr1)
      z1=iand(z1,3)
c      z1=ishft(z1,-2)
      call output (adr2,let2)
      z2=input(adr1)
      zi(5)=z1+(z2*4)
      if (e.eq.0) go to 630
      if ((pi(5)-zi(5)).eq.0) go to 600
      if ((zi(5)-pi(5)).lt.0) go to 610
      go to 630
600  ab=ab-16
      call outport (ab,amot)
      call beep2(fre,dur,in)
c      ---- stampa su schermo pos.raggiunta----
      iseg=45056
      i=900
      jp=ichar(as$)
      call poke (iseg,i,jp)
      e=0
      go to 630
610  ad=ad-16
630  continue
c      ---- Gestione Canale 6 ----
```

```
val=5
call outport (val,mux)

c      ---- start conversion ----

val=1
call outport (val,sconv)

c      ---- lettura flag ----
632    call output (adr2,flag)
val=input (adr1)
if (val.eq.1) go to 632
c      ---- lettura posizione vaso ----
call output (adr2,let1)
z1=input (adr1)
z1=iand(z1,3)
c      z1=ishft(z1,-2)
call output (adr2,let2)
z2=input (adr1)
zi(6)=z1+(z2*4)
if (f.eq.0) go to 680
if ((pi(6)-zi(6)).eq.0) go to 650
if ((zi(6)-pi(6)).lt.0) go to 660
go to 680
650    ab=ab-32
call outport (ab,amot)
call beep2 (fre,dur,in)
c      ---- stampa su schermo pos. raggiunta----
iseg=45056
i=914
jp=ichar(as$)
call poke(iseg,i,jp)
f=0
go to 680
660    ad=ad-32
680    continue
c      ---- Gestione Canale 7 ----

val=6
call outport (val,mux)

c      ---- start conversion ----

val=1
call outport (val,sconv)

c      ---- lettura flag ----
682    call output (adr2,flag)
val=input (adr1)
if (val.eq.1) go to 682
```

```
c      ---- lettura posizione vaso ----
      call output (adr2,let1)
      z1=input (adr1)
      z1=iand(z1,3)
c      z1=ishft(z1,-2)
      call output (adr2,let2)
      z2=input (adr1)
      zi(7)=z1+(z2*4)
      if (g.eq.0) go to 730
      if ((pi(7)-zi(7)).eq.0) go to 700
      if ((zi(7)-pi(7)).lt.0) go to 710
      go to 730
700    ab=ab-64
      call outport (ab,amot)
      call beep2 (fre,dur,in)
c      ---- stampa su schermo pos. raggiunta----
      iseg=45056
      i=928
      jp=ichar(as$)
      call poke (iseg,i,jp)
      g=0
      go to 730
710    ad=ad-64
730    continue
c      ---- Gestione Canale 8----
      val=7
      call outport (val,mux)

c      ---- start conversion ----
      val=1
      call outport (val,sconv)

c      732   ----lettura flag ---
      call output (adr2,flag)
      val=input (adr1)
      if (val.eq.1) go to 732
c      ---- lettura posizione vaso ----
      call output (adr2,let1)
      z1=input (adr1)
      z1=iand(z1,3)
c      z1=ishft(z1,-2)
      call output (adr2,let2)
      z2=input (adr1)
      zi(8)=z1+(z2*4)
      if (h.eq.0) go to 780
      if ((pi(8)-zi(8)).eq.0) go to 750
      if ((zi(8)-pi(8)).lt.0) go to 760
      go to 780
750    ab=ab-128
      call outport (ab,amot)
      call beep2(fre,dur,in)
```

```
c      ---- stampa su schermo pos. raggiunta----
c      iseg=45056
c      i=942
c      jp=ichar(as$)
c      call poke (iseg,i,jp)
c      h=0
c      go to 780
760    ad=ad-128

c      =====
c      movimento/blocco motori
c      =====

780    call outport (ad,smot)
c      call outport (ab,amot)

c      -----
c      ind=ind+1
c      if (ab.eq.0) go to 850
c      if (ind.eq.2800)go to 830
820    ad=255
c      ---- stampa lettura canali -----
c      do 823 i=1,8
c      z(i)=float(zi(i))*0.0980391
823    continue
c      write (out,825) z(1),z(2),z(3),z(4),z(5),z(6),z(7),z(8)
825    format('Posizione corrente',2x,f4.1,3x,f4.1,3x,f4.1,
c      ,3x,f4.1,3x,f4.1,3x,f4.1,3x,f4.1)
c      iseg=45056
c      j=0
c      do 828 i=640,799,2
c      j=j+1
c      jp=ichar(chr(j))
828    call poke (iseg,i,jp)
c      ---- procedura di controllo di 'USCITA' programma -----
c      irow=23
c      icolm=23
c      ivpage=0
c      call putcur(irow,icollm,ivpage,ierror)
c      write(*,*)'          PER USCIRE DURANTE IL MOVIMENTO VASI -->ESC'
c      call inkey(cr,key,in)
c      if(key.eq.1) then
c      =====
c      Reset di tutto il Sistema
c      =====
c      val=0
c      call output (adr4,val)
c      ---- scrittura su disco 99 99=punt.NS interrotto!!'-----
c      write (10,829,rec=1)99
c      write (10,829,rec=2)99
829    format(2x,i2)
c      stop'                                fine programma!!!
c      endif
c      go to 380
```

```
830    continue
      val=0
      call output (val,amot)
      do 840 i=1,4000000
      del=cos (ze)
840    continue
      call outport (ab,amot)
      ind=1
      go to 820
850    continue
c     ---- lettura di controllo a vasi fermi ----
      data m(1)/0/,m(2)/0/,m(3)/0/,m(4)/0/,m(5)/0/,m(6)/0/,m(7)/0/
      : ,m(7)/0/,m(8)/0/
c     ---- interv.di assestamento vasi----
      call beep2 (fre,dur,in)
      do 851 i=1,10000
851    continue
c     ---- 100 letture ---
      do 870 j=1,100
      do 860 i=1,8

      val=i-1
      call outport (val,mux)

c     ---- start conversion ----

      val=1
      call outport (val,sconv)

c     ---- lettura flag ----
855    call output (adr2,flag)
      val=input (adr1)
      if (val.eq.1) go to 855
c     ---- lettura A/D ---
      call output (adr2,let1)
      z1=input (adr1)
      call output (adr2,let2)
      z2=input (adr1)
      zi(i)=z1+(z2*4)
      z(i)=float(zi(i))*0.0980391
      m(i)=m(i)+z(i)
860    continue
870    continue
      do 890 i=1,8
890    z(i)=m(i)/100
c     ---- riposizionamento del cursore ----
      irow=6
      icolm=1
      ivpage=0
      call putcur(irow,icolm,ivpage,ierror)
c     ---- calcolo e stampa scarti ----
      do 900 i=1,8
      z(i)=(z(i)-l(i))*10
```

```
900      zs(i)=nint(z(i))
910      write (*,910)zs(1),zs(2),zs(3),zs(4),zs(5),zs(6),zs(7),zs(8)
         format (' Scarti (mm)',9x,i3,4x,i3,4x,i3,4x,i3,4x,i3,
         : 4x,i3,4x,i3,4x,i3)
c       ---- stampa della 910 ----
         iseg=45056
         j=15
         do 920 i=961,1119,2
920      call poke (iseg,i,j)
         write (*,*)'RITARDI:INSERITI! | PROVE B/E:OK! | PUNTAMENTO:'
         : ,punt$,'|',' FINE CORSA:'
         iseg=45056
         j=112
         do 925 i=1121,1279,2
925      call poke(iseg,i,j)

c       --- FORMAZIONE FASCIO NS ---
call fasens(dg,pr)

      stop
1000     fre=1500
         dur=15
         call beep2(fre,dur,in)
         call cls
         fre=800
         dur=10
         punt$='MANUALE'
         call beep2(fre,dur,in)
         write (*,*)'PUNTAMENTO MANUALE DEI VASI'
         write (*,*)'-----'
         write (*,*)'
         write (*,*)'STESSO LIVELLO-----(1)'
         write (*,*)'LIVELLI DIVERSI-----(2)'
         read (*,*) liv
         if (liv.eq.1)go to 1100
         if (liv.eq.2)go to 1500
         go to 1000
c       ---- livello manuale oli ----
1100     liv=0
         write (*,*)'Entrare il livello (cm) a cui portare gli 8 Vasi'
         read (*,*) liv
         if (liv.gt.86.1.or.liv.le.14.99) go to 1000
         liv=liv*10.2
         do 1110 i=1,8
         pi(i)=liv
1110     continue
         go to 370
c       ---- input livelli "manuali" -----
1500     call cls
         liv=0
         fre=2000
         dur=4
         call beep2(fre,dur,in)
         write (*,*) 'Dare il Livello di ogni Vaso (in cm)'
1550     write (*,*)'Livello vaso 1 ?'
```

```
    read (*,*) liv
    if (liv.le.14.99.or.liv.gt.86.1) go to 1550
    liv=liv*10.2
    pi(1)=liv
    liv=0
1560  write (*,*)"Livello Vaso 2 ? "
    read (*,*) liv
    if (liv.le.14.99.or.liv.gt.86.1) go to 1560
    liv=liv*10.2
    pi(2)=liv
    liv=0
1570  write (*,*)"Livello vaso 3 ? "
    read (*,*) liv
    if (liv.le.14.99.or.liv.gt.86.1) go to 1570
    liv=liv*10.2
    pi(3)=liv
    liv=0
1580  write (*,*)"Livello Vaso 4 ?"
    read (*,*) liv
    if(liv.le.14.99.or.liv.gt.86.1) go to 1580
    liv=liv*10.2
    pi(4)=liv
    liv=0
1590  write (*,*)"Livello vaso 5 ?"
    read (*,*) liv
    if (liv.le.14.99.or.liv.gt.86.1) go to 1590
    liv=liv*10.2
    pi(5)=liv
    liv=0
2000  write (*,*)"Livello Vaso 6 ?"
    read (*,*) liv
    if (liv.le.14.99.or.liv.gt.86) go to 2000
    liv=liv*10.2
    pi(6)=liv
    liv=0
2010  write (*,*)"Livello Vaso 7?"
    read (*,*) liv
    if (liv.le.14.99.or.liv.gt.86.1) go to 2010
    liv=liv*10.2
    pi(7)=liv
    liv=0
2020  write (*,*)"Livello Vaso 8?"
    read (*,*) liv
    if (liv.le.14.99.or.liv.gt.86.1) go to 2020
    liv=liv*10.2
    pi(8)=liv
    liv=0
    go to 370
end
```

```
$DECLARE
c -----
c SUBROUTINE PER LA FORMAZIONE DEL FASCIO NS
c tenendo conto dell'offset di fase EW-NS
c (modificare quando cambia!)
c -----
c nome= FASENS

subroutine fasens(dg,pr)
real d1(8),ze,dc,df(8),uno
integer*2 val,i,outp(8),res,of
real dg,pr,ff,fase,off
ze=44.57
c ----- OFFSET EW-NS -----
off=113
c -----
uno=1
res=0
ff=0
fase=0
write(*,*)'
write(*,*)'
write(*,*)'Fascio NS Formato !'
write(*,100) off
100 format(' Offset di fase EW/NS Inserito (gradi) =',f4.0)
dc=dg
dc=dc+pr/60.
do 30 i=1,8
d1(i)=(i-4.5)*80*sin((ze-dc)*3.141592/180.)
d1(i)=d1(i)*408./299.793
val=int(d1(i))
df(i)=d1(i)-val
df(i)=df(i)*360.
30 continue
do 61 i=1,8
if(df(i).lt.0) df(i)=df(i)+360.
outp(i)=nint(df(i)/(360./32.))
if(outp(i).ge.32) outp(i)=outp(i)-32
61 continue
do 1000 i=1,8

c ----- introduzione offset e gestione sfasatori-----
of=nint(off/(360./32.))
outp(i)=outp(i)+of
if(outp(i).ge.32) outp(i)=outp(i)-32
if(outp(i).lt.0) outp(i)=outp(i)+32
call outsfns(i-1,outp(i))
1000 continue
return
end
```

```
subroutine outsfns(roma_a,roma_d)

integer*2 roma_a,roma_d
integer*2 port_a,port_d,strobe,ifreq,idurr,in
ifreq=1300
idurr=5
call beep2(ifreq,idurr,in)

100 continue
port_a=82
port_d=81
strobe=roma_a+128
call outport(roma_d,port_d)
call outport(roma_a,port_a)
call outport(strobe,port_a)
call outport(roma_a,port_a)
return
end
```

\$DECLARE

```
subroutine outport(data,address)
integer*2 adr1,adr2,adr3,w,r,address,data
adr1=768
adr2=769
adr3=770
w=0
r=1
call output (adr2,address)
call output(adr1,data)
call output (adr3,w)
call output(adr3,r)
return
end
```

## **BIBLIOGRAFIA**

- PC 488 Programming and Reference Manual and Software.  
CEC Capital Equipment Corporation

**NOTE**