PPC Journal

Personal Programming Center

V 14 N6

June 1987

The Persona Programming Center is an interational users group of People Programming Computers The Original Hewlett-Packard users

Contents

PPC, the Persoal Programming Center, was a California non-profit public benefit corporation dedicated to advancing the applications art of personal computing. PPC is the oldest personal computing users group and published the PPC Journal. Address no correspondence to PPC, as the organization is presently disbanded. However, we do continue informal monthly meetings. Inquire by e-mail to rnelson@educalc.com

GRAPHING OF SECOND DEGREE CONICS
Part IV

By Ted Levine #12360
The Villages 89D
Freehold, NJ 07728

In this segment we shall give some examples of Degenerate Conics which consist of points, parallel lines, intersecting lines and no loci. Also included are several new programs and a few revised programs which will make for more efficient problem solving and drawing of Second Degree Conics.

SUMMARY

$AX^2+BXY+CY^2+DX+EY+F=0$ General Form 2nd Degree Conic
$AX^2+CY^2+DX+EY+F=0$ No BXY term
$AX^2+DX+EY+F=0$ Parabola $X^2=2pY$
$CY^2+DX+EY+F=0$ Parabola $Y^2=2pX$
$AX^2+BXY+CY^2+F=0$ No DX or EY terms,h=0,k=0 No translation.
$AX^2-CY^2=0$ X Lines(intersecting lines)
$-AX^2-CY^2-F=0$ No locus
$AX^2+CY^2=0$ Point
$AX^2+DX+F=0$ Quadratic(Parallel Lines)

NO BEEP
1.Enter values
2.XEQ"CONIC"
3.Translation
a)If BXY term use matrix form to calculate h and k.(If no DX and no EY, XEQ"RMVXY")
b)If no BXY term but DX and EY terms XEQ"TRANS".
c)XEQ"SUM"
4.XEQ"RMVXY"(If BXY term)
5.XEQ"STAFOR"
a)Calculate points on curve.
6.XEQ"ROTATE"(If BXY term)
XEQ"INVT"(If no BXY term)
7.Use values X and Y from step 6 to check in equation I.

BEEP

1.Enter values.
2.XEQ"CONIC"
3.XEQ"RMVXY"(If BXY term)
XEQ"TRANS"(If no BXY term)
Reduce A'X² to A'=1 by dividing equation by A'.
4.Form STANDARD FORM. Calculate points on curve.
5.XEQ"INVT"
6.XEQ"ROT"(If BXY term)
7.Use values X and Y from step 6 to check in equation I.

EXAMPLE 1

I $X^2+4XY+4Y^2-8=0$

DO	SEE
XEQ"CLRG"	
1 STO 18	18
4 STO 19	19
4 STO 20	20
8 CHS STO 23	-8
XEQ"CONIC"	-160(PSE)BEEP - 2 parallel lines
XEQ"RMVXY"	-1.3333(PSE) .6000
CHS	-.6000
R/S	0
RCL 30	63.43495
RCL 34	5
RCL 35	0
RCL 36	0
RCL 37	0
RCL 38	0
RCL 23	-8

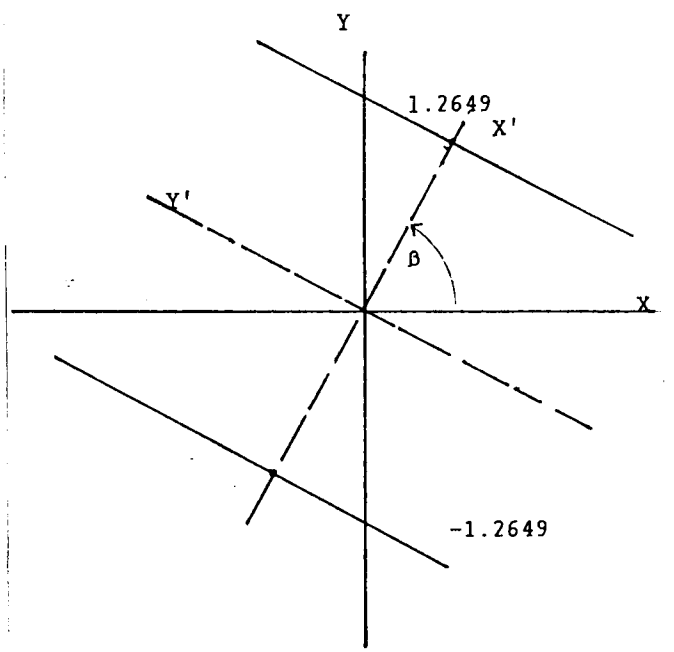
II $5(X')^2=8$
Since there are no DX and EY terms,no translation,that is,h=0,k=0.
IIa X'=±1.2649 STO 40

XEQ"ROTATE"	X= .56568
R/S	Y= 1.13136

Substitute X and Y in equation I for check.

EXAMPLE 1

I $X^2+4XY+4Y^2-8=0$



$5(X')^2=8$

EXAMPLE 2

I $X^2+4XY+4Y^2+4X+8Y+3=0$

DO	SEE
XEQ"CLRG"	
1 STO 18	1
4 STO 19	4

While the Personal Programming Center (PPC), publishers of the PPC Journal, is operated on a volunteer basis, the opinions, insights and predictions of the author's contributing articles to be published in the PPC Journal do not necessarily represent the views and opinions of PPC. If you have any questions or comments, send your correspondence to PPC, P.O. Box 90579, Long Beach, California 90809-0579.

STO 20	4
STO 21	4
8 STO 22	8
3 STO 23	3
XEQ"CONIC"	-20(PSE)BEEP 2 parallel lines
XEQ"RMVXY"	-1.3333(PSE) .6000
CHS	-.6000
R/S	0
RCL 30	63.43495
RCL 34	5
RCL 35	0
RCL 36	0
RCL 37	8.94427
RCL 38	0
RCL 23	3

II $5(X')^2+8.94427X'+3=0$

This is in quadratic form.

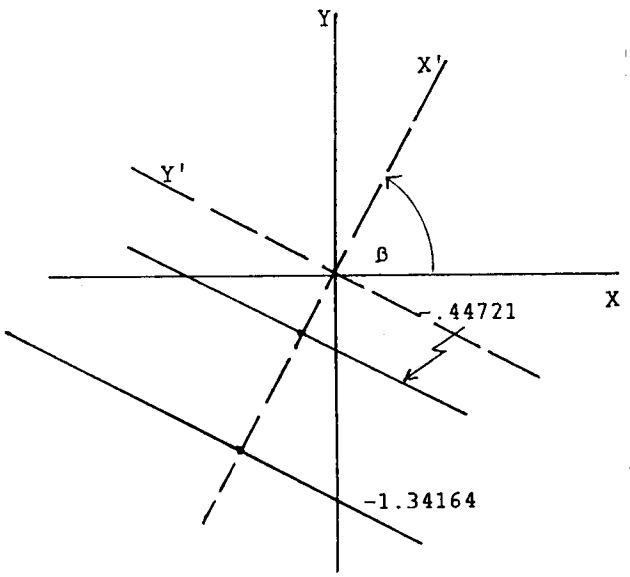
	(†)
XEQ"QE"	-.44721
R/S	ROOT 1=-.44721
R/S	ROOT 2=-1.34164
ROOT 1 STO 40	
XEQ"ROT"	X=-.2
R/S	Y=-.4
ROOT 2 STO 40	
XEQ"ROT"	X=-.6
R/S	Y=-1.2

Substitute X and Y into I for check.

(†) RCL 34 ENTER↑
RCL 37 ENTER↑
RCL 23
XEQ"QE"

EXAMPLE 2

I $X^2+4XY+4Y^2+4X+8Y+3=0$



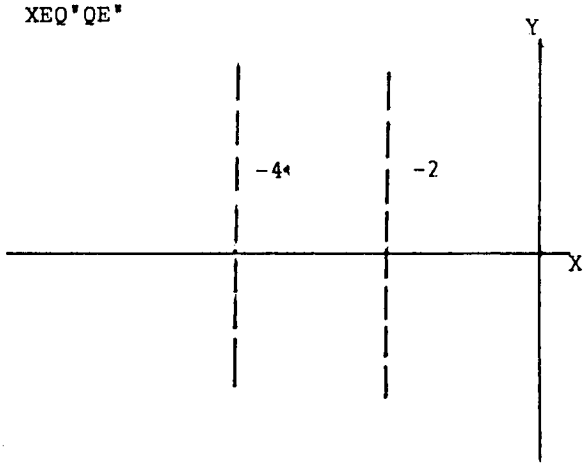
$5(X')^2+8.94427X'+3=0$

EXAMPLE 3

I $X^2+6X+8=0$

DO	SEE
XEQ"CLRG"	
1 STO 18	1
6 STO 21	6
8 STO 23	8
XEQ"CONIC"	-4(PSE)BEEP 2 parallel lines
I is in quadratic form	(†)
XEQ"QE"	-2
R/S	ROOT 1=-2
R/S	ROOT 2=-4

(†) RCL 18 ENTER↑
RCL 21 ENTER↑
RCL 23
XEQ"QE"



EXAMPLE 4

I $X^2-Y^2-6X-10Y-16=0$

DO	SEE
XEQ"CLRG"	
! STO 18	1
1 CHS STO 20	-1
6 CHS STO 21	-6
10 CHS STO 22	-10
16 CHS STO 23	-16
XEQ"CONIC"	X LINES NO BEEP (intersecting lines)

Since there is no BXY term,XEQ"TRANS"

XEQ"TRANS"	H=3 K=-5
XEQ"SUM"	0

Form for intersecting lines

RCL 18	1
RCL 20	-1
RCL 23	0

II $(X')^2-(Y')^2=0$

IIa $Y'=\pm X'$

Standard form for line $Y=mX+b$

From IIa m=1 and m=-1

Inverse tan 1=45° and -1=135°

If X'=1 then Y'=1

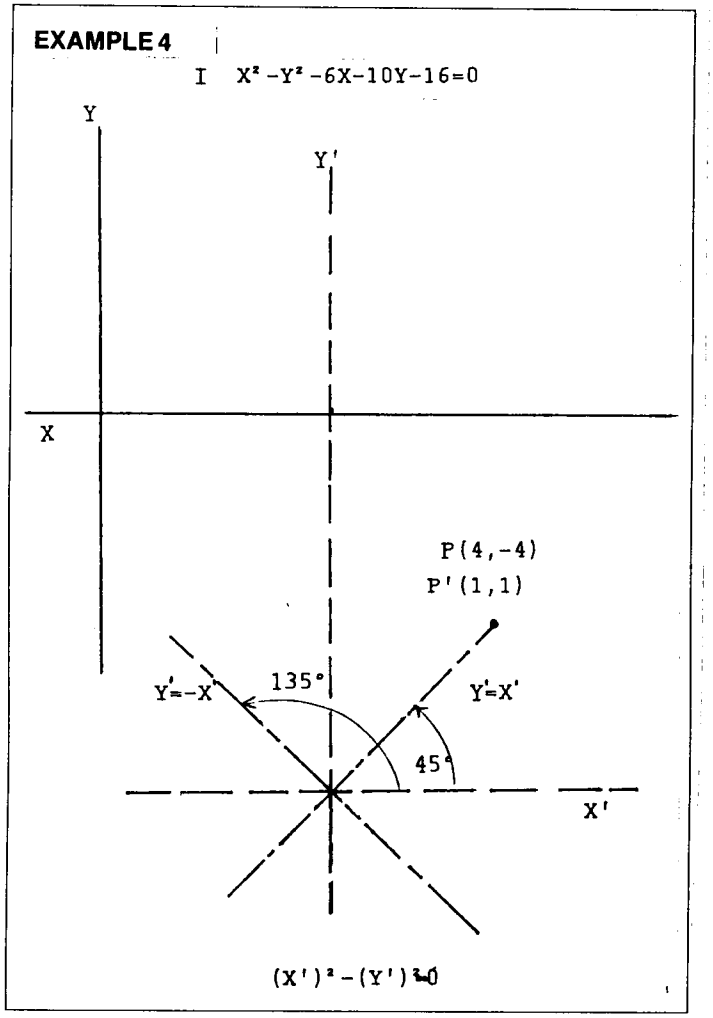
X' STO 40
Y' STO 41

XEQ"INVT"	X=4
R/S	Y=-4

Substituite X and Y into I for check.

EXAMPLE 4

I $X^2-Y^2-6X-10Y-16=0$



EXAMPLE 5

I $X^2+XY-6Y^2=0$

DO	SEE
XEQ"CLRG"	
1 STO 18	1
1 STO 19	1
6 CHS STO 20	-6
XEQ"CONIC"	X LINES NO BEEP (intersecting lines)

No DX and no EY terms,XEQ"RMVXY"

XEQ"RMVXY"	.14286(PSE) .98995
R/S	0
RCL 30	4.06505
RCL 34	1.03553
RCL 35	0
RCL 36	-6.03553
RCL 37	0
RCL 38	0
RCL 23	0

II $1.03553(X')^2-6.03553(Y')^2=0$

IIa $Y'=\pm.41421X'$

This is the standard form for a line.

Inverse tan .41421=22.5°,-.41421=157.5°

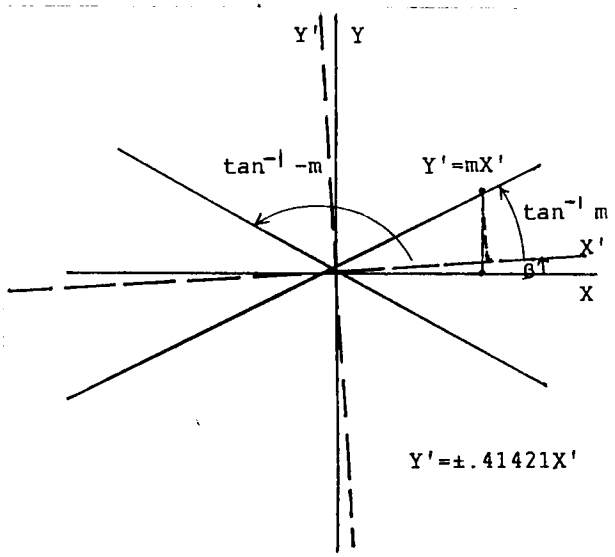
If X'=1 STO 40 then Y'=.41421 STO 41

XEQ"X LINES"	X=.96812
R/S	Y=.48406

Substitute X and Y into I for check.

EXAMPLE 5

I $X^2+XY-6Y^2=0$



EXAMPLE 6

I $9X^2+24XY+16Y^2-36X-48Y+61=0$

DO	SEE
XEQ"CLRG"	
9 STO 18	9
24 STO 19	24
16 STO 20	16
36 CHS STO 21	-36
48 CHS STO 22	-48
61 STO 23	61
XEQ"CONIC"	+2500 BEEP + NO LOCUS
XEQ"RMVXY"	-3.42857(PSE) .2800
CHS	-.2800
R/S	0
RCL 30	53.1301
RCL 34	25
RCL 35	0
RCL 36	0
RCL 37	-60
RCL 38	0
RCL 23	61

II $25(X')^2-60X'+61=0$

This is in quadratic form (†)

XEQ"QE"	1.2
R/S	ROOT 1=1.2±1i

No real root thus no locus.

(†) RCL 34 ENTER↑
RCL 37 ENTER↑
RCL 23
XEQ"QE"

EXAMPLE 7

I $X^2+Y^2+4X-6Y+20=0$

DO	SEE
XEQ"CLRG"	
1 STO 18	1
1 STO 20	1
4 STO 21	4
6 CHS STO 22	-6
20 STO 23	20
XEQ"CONIC"	+14 NO BEEP NO LOCUS
XEQ"TRANS"	H=-2 K=3
XEQ"SUM"	7
XEQ"STAFOR"	-7
R/S	2.64575
R/s	-7
R/S	2.64575

II $-(X')^2/7-(Y')^2/7=1$

Since the sum of negatives is not positive, there is no locus for a real value.

EXAMPLE 8

I $14X^2-4XY+11Y^2-88X+34Y+149=0$

DO	SEE
XEQ"CLRG"	
14 STO 18	14
4 CHS STO 19	-4
11 STO 20	11
88 CHS STO 21	-88
34 STO 22	34
149 STO 23	149
XEQ"CONIC"	POINT NO BEEP
XEQ"MIO" $\begin{bmatrix} 28 & -4 & -88 \\ -4 & 22 & 34 \end{bmatrix}$	H=3
XEQ"RRM"	
GTO"MIO"	K=-1
USER B	
3 STO 16	
1 CHS STO 17	
XEQ"SUM"	0
XEQ"RMVXY"	-1.3333(PSE)
	.6000
CHS	-.6000
R/S	0
RCL 30	63.43495
RCL 34	10
RCL 35	0
RCL 36	15
RCL 37	0
RCL 38	0
RCL 23	0

II $10(X')^2+15(Y')^2=0$

X' and Y' must each be =0 to satisfy II.

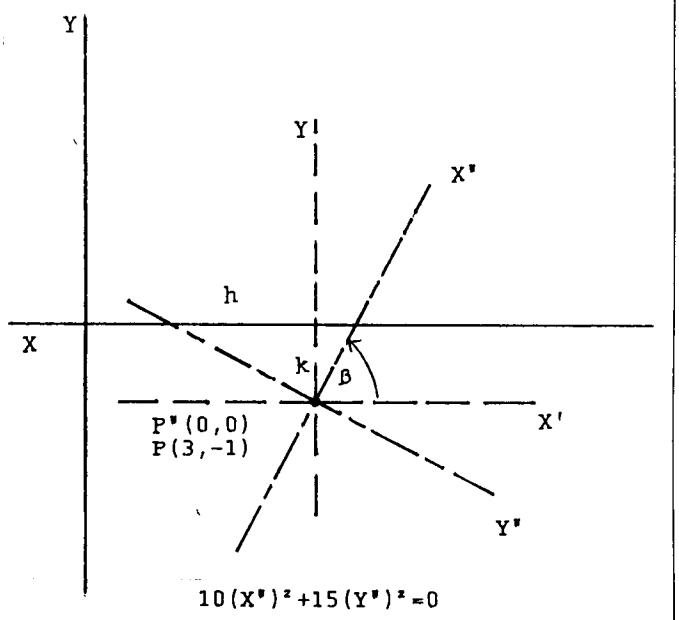
0 STO40
0 STO 41

XEQ"ROTATE"	X=3
R/S	Y=-1

Substitute X and Y into I for check.

EXAMPLE 8

I $14X^2-4XY+11Y^2-88X+34Y+149=0$



EXAMPLE 9

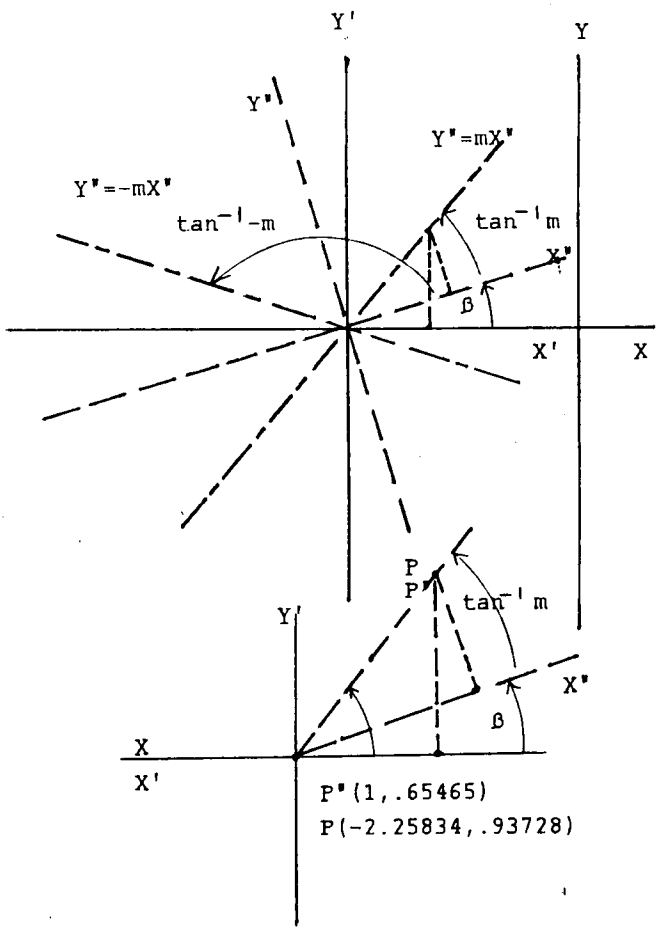
I $X^2+3XY-3Y^2+6X+9Y+9=0$

DO	SEE
XEQ"CLRG"	
1 STO 18	1
3 STO 19	3
3 CHS STO 20	-3
6 STO 21	6
9 STO 22	9
STO 23	9
XEQ"CONIC"	X LINES NO BEEP
XEQ"MIO" $\begin{bmatrix} 2 & 3 & 6 \\ 3 & -6 & 9 \end{bmatrix}$	H=-3
XEQ"RRM"	
GTO"MIO"	K=0
USER B	
-3 STO 16	-3
0 STO 17	0
XEQ"SUM"	0
XEQ"RMVXY"	.7500(PSE)
	.8000
R/S	0
RCL 30	18.43495
RCL 34	1.5
RCL 35	0
RCL 36	-3.5
RCL 37	0

```
RCL 38                    0
RCL 23                    0

II  1.5(X')²-3.5(Y')²=0
IIa  Y'=±.65465X'

This is standard form for a line.
Inverse tan .65465=33.21076°
Inverse tan-.65465=146.78924°
If X'=1 STO 40 then Y'=.65465 STO 41

XEQ"X LINES"              X=-2.25834
                          Y=.93728
```

EXAMPLE 9

I $X^2+3XY-3Y^2+6X+9Y+9=0$



Y"=±.65465X"

```
01 LBL"SUM"
02 RCL 18
03 RCL 16
04 X↗2
05 *
06 RCL 19
07 RCL 16
08 *
09 RCL 17
10 *
11 +
12 RCL 20
13 RCL 17
14 X↗2
15 *
16 +
17 RCL 21
18 RCL 16
19 *
20 +
21 RCL 22
22 RCL 17
23 *
24 +
25 ST+23
26 0
27 STO 21
28 0
29 STO 22
30 RCL 19
31 X=0?
32 GTO 01
33 RCL 23
34 LBL 01
35 RCL 18
36 STO 34
37 RCL 20
38 STO 36
39 RCL 23
40 END
```

```
01 LBL"X LINES"
02 RCL 41
03 ENTER↗
04 RCL 40
05 R-P
06 STO 42
07 X<>Y
08 RCL 30
09 +
10 STO 30
11 RCL 30
12 ENTER↗
13 RCL 42
14 P-R
15 STO 40
16 X<>Y
17 STO 41
18 XEQ"INVT"
19 END
```

```
01 LBL"ROTATE
02 RCL 40
03 RCL 32
04 *
05 RCL 41
06 RCL 29
07 *
08 -
09 RCL 16
10 +
11 STO 01
12 CLA
13 "├ X="
14 ARCL 01
15 AVIEW
16 STOP
17 RCL 40
18 RCL 29
19 *
20 RCL 41
21 RCL 32
22 *
23 +
24 RCL 17
25 +
26 STO 02
27 CLA
28 "├ Y="
29 ARCL 02
30 AVIEW
31 END
```

```
01 LBL"ROT"
02 RCL 40
03 RCL 32
04 *
05 RCL 41
06 RCL 29
07 *
08 -
09 STO 01
10 CLA
11 "├ X="
12 ARCL 01
13 AVIEW
14 STOP
15 RCL 40
16 RCL 29
17 *
18 RCL 41
19 RCL 32
20 *
21 +
22 STO 02
23 CLA
24 "├ Y="
25 ARCL 02
26 AVIEW
27 END
```

```
01 LBL"QE"
02 CF 04
03 X<>Z
04 ST/Z
05 ST+X
06 /
07 CHS
08 ENTER↗
09 ENTER↗
10 X↗2
11 R↗
12-
13 X<0?
14 SF 04
15 ABS
16 SQRT
17 ST-Z
18 X<>Y
19 FC? 04
20 +
21 RTN
22 SF 21
23 FS?C 04
24 GTO 14
25 "ROOT 1="
26 ARCL X
27 AVIEW
28 "ROOT 2="
29 ARCL Y
30 AVIEW
31 RTN
32 LBL 14
33 CLA
34 ARCL X
35 "├ +-"
36 ARCL Y
37 "├ I"
38 AVIEW
39 END
(Robert Groom
   5127)
AX²+DX+F=0

A ENTER↗
D ENTER↗
F
XEQ"QE"
```

```
01 LBL "TRANS"
02 RCL 18
03 X=0?
04 GTO "A"
05 RCL 20
06 X=0?
07 GTO "B"
08 XEQ"TRA"
09 LBL"A"
10 RCL 22
11 RCL 20
12 2
13 *
14 /
15 CHS
16 STO 01
17 STO 17
18 CLA
19 "├ k="
20 ARCL 01
21 AVIEW
22 PSE
23 PSE
24 X↗2
25 CHS
26 STO 02
27 RCL 23
28 RCL 02
29 +
30 RCL 21
31 /
32 CHS
33 STO 03
34 STO 16
35 CLA
36 "├ H="
37 ARCL 03
38 AVIEW
39 STOP
40 LBL"B"
41 RCL 21
42 RCL 18
43 2
44 *
45 /
46 CHS
47 STO 01
48 STO 16
49 CLA
50 "├ H="
51 ARCL 01
52 AVIEW
53 PSE
54 PSE
55 X↗2
56 CHS
57 STO 02
58 RCL 23
59 RCL 02
60 +
61 RCL 22
62 /
63 CHS
64 STO 03
65 STO 17
66 CLA
67 "├ k="
68 ARCL 03
69 AVIEW
70 STOP
71 END
```

```
01 LBL"INVT"
02 RCL 40
03 RCL 16
04 +
05 STO 40
06 CLA
07 "├ X="
08 ARCL 40
09 AVIEW
10 R/S
11 RCL 41
12 RCL 17
13 +
14 STO 41
15 CLA
16 "├ Y="
17 ARCL 41
18 AVIEW
19 END
```

```
01 LBL"TRA"          20 /
02 RCL 21            21 CHS
03 RCL 18            22 STO 02
04 2                 23 STO 17
05 *                 24 CLA
06 /                 25 "├ k="
07 CHS               26 ARCL 02
08 STO 01            27 AVIEW
09 STO 16            28 STOP
10 CLA               29 END
11 "├ H="
12 ARCL 01
13 AVIEW             AX² +CY² +DX+EY+F=0
14 PSE
15 PSE               A STO 18
16 RCL 22            C STO 20
17 RCL 20            D STO 21
18 2                 E STO 22
19 *                 F STO 23
```

R/S

PROGRAM C/F CALENDAR FUN

By Samuel J. Hartman #12556
5954 Colfax Avenue Apt. #5
N. Hollywood, CA 91601

By Spencer J. Hartman #9541
2432 Stow Street
Simi, CA 93063-3558

This keen little program can be used for determining the days between two dates in years, months and days . . . It will print out the day of the week for any selected date . . . and it also will calculate the number of years, months, and days in a specified number of days.

The program is useful in that it will determine the age of a parent, for example, when his children were born. Enter the two dates and the age is printed out. Or, it can determine the time span between two birthdays . . . in years, months, days.

How many times have we asked ourselves, "How old was I when junior was born?" Or, "How many years separate Mary and Tom?"

Most programs of this type are called Calendar Functions. We have named ours Calendar Fun!

To run the program, XEQ C/F and answer prompts for LATEST YEAR?, MONTH?, and DAY? Press R/S after each entry. Answer prompts for 2nd YR?, MONTH?, and DAY? Press R/S after each entry. Program will print out the number of days between dates, also the number of years, months and days. Press R/S to repeat or USER A.

Label B. Press USER B. A data chart of corresponding numbers and days is printed out. Answer prompts for YEAR?, MONTH, and DAY? The date is printed out and a number which corresponds with the day of the week. i/e 5 = Friday 0 = Sunday etc.

Label C. Press USER C and answer prompt for # OF DAYS? The program prints out the number of Years, Months, and Days in that time span.

Great for determining the present ages of family members, differences in years, months, and days between ages etc. Also the day of the week born, or anniversaries and so on.

SAMPLE PRINT OUTS

```
DAYS BET. DATES                DAYS BET. DATES
5/13/1987      9/6/1915        2/11/1940      9/6/1915
# OF DAYS       26,182.        # OF DAYS        8,924.
 71.Y 8.M 5.D  .                24.Y 5.M 6.D

0 1 2 3 4 5 6                  0 1 2 3 4 5 6
S M T W T F S                  S M T W T F S
DATE           9/6/1915        DATE           2/11/1940
DAY OF WEEK          1.        DAY OF WEEK          0.

# OF DAYS?      26,182.        # OF DAYS?      999,999.
 71.Y 8.M 5.D                   2,737.Y 9.M 18.D
```

```
01*LBL "C/F"
02*LBL A
03 CF 27
04 "DAYS BET. DATES"
05 AVIEW
06 FIX 0
07 SF 00
08 "1ST YR?"
09 XEQ 01
10 "2ND YR?"
11 XEQ 01
12 RCL 01
13 X<>Y
14 -
15 "# OF DAYS"
16 ACA
17 FMT
18 ACX
19 PRBUF
20 GTO 02
21*LBL B
22 CF 27
23 FIX 0
24 CLA
25 "0 1 2 3 4 5 6"
26 PRA
27 "S M T W T F S"
28 PRA
29*LBL 00
30 "DATE"
31 ACA
32 FMT
33 "YEAR?"
34 XEQ 01
35 1
36 +
37 7
38 MOD
39 "DAY OF WEEK"
40 ACA
41 FMT
42 ACX
43 ADV
44 ADV
45 STOP
46 GTO 00
47*LBL C
48 CF 27
49 "# OF DAYS?"
50 ACA
51 FMT
52 PROMPT
53 ACX
54 PRBUF
55 GTO 02
56*LBL 01
57 XROM "T1"
58 PROMPT
59 CF 28
60 CF 29
61 "MONTH"
62 PROMPT
63 "DAY"
64 PROMPT
65 CLA
66 ARCL Y
67 "├/"
68 ARCL X
69 "├/"
70 ARCL Z
71 ACA
72 FMT
73 FC? 00
74 PRBUF
75 XROM "CJ"
76 FS?C 00
77 STO 01
78 SF 28
79 SF 29
80 RTN
81*LBL 02
82 CF 27
83 365.2564
84 /
85 FIX 1
86 INT
87 LASTX
88 FRC
89 365.2564
90 *
91 30.44
92 /
93 INT
94 LASTX
95 FRC
96 30.44
97 *
98 " "
99 FIX 0
100 ARCL Z
101 "├Y "
102 ARCL Y
103 "├M "
104 ARCL X
105 "├D "
106 PRA
107 ADV
108 STOP
109 GTO A
110 .END.
```

Needed: Printer and PPC Rom

C/F

REGISTERS: 45

ROW 1 (1-4)

ROW 2 (4-4)

ROW 3 (4-8)

ROW 4 (8-10)

ROW 5 (11-15)

ROW 6 (15-20)

ROW 7 (21-25)

ROW 8 (25-27)

ROW 9 (27-28)

ROW 10 (29-33)

ROW 11 (33-39)

ROW 12 (39-41)

ROW 13 (42-49)

ROW 14 (49-51)

ROW 15 (52-59)

ROW 16 (60-63)

ROW 17 (64-70)

ROW 18 (70-76)

ROW 19 (77-83)

ROW 20 (83-89)

ROW 21 (89-93)

ROW 22 (94-100)

ROW 23 (100-104)

ROW 24 (105-110)

ROW 25 (110-110)

R/S

BITS & PIECES

Here is a short program I wrote to quickly change the contrast of the HP-71B depending on the viewing angle needed. I use the 71 in many different situations and this program makes it easy to change the contrast from desktop to lap viewing. Run the program from basic; the screen will clear to display CONTRAST = n, where n is the current contrast. Press the "+" or "-" keys to adjust the contrast up or down to the desired viewing angle. The default contrast is saved in line 40, to reset the contrast to the default position, press "R". When you have the proper angle selected, press "E" to exit the program.

```
10  ! CONTRAST ADJUSTMENT
20  ! By John M. Wargo  (11826)
30  DIM A$,C$, @ INTEGER C
40  C1=HTD(PEEK$("2E3FE",1)) @
     ! C1 = starting CONTRAST
50  C=C1 @ C$="Contrast = " @ FIX 0
60  CONTRAST C @ DISP C$,C
70  A$=KEY$
80  IF A$="+" AND C=15 THEN 70 ELSE IF
     A$="+" THEN C=C+1 @ GOTO 60
90  IF A$="-" AND C=0  THEN 70 ELSE IF
     A$="-" THEN C=C-1 @  GOTO 60
100 IF A$="R" THEN C=C1 @ GOTO 60
110 IF A$="E" THEN 120 ELSE 70
120 FIX 4 @ PUT "#43" @ END
```

I also have a question that I hope you or some of the PPC members can answer for me. The other day I turned on Fred, my HP-41CV, and I couldn't get it to work. Every time I pressed a key, or tried to execute a program or function, the 41 would emit a continuous tone while it was working. I tried many things to get it to stop doing this, even tried to reset the memory, but none of this would work. I eventually had to take the batteries out and let it sit for a while. When I put the batteries back in the next day, everything worked fine. Can any of you tell me what happened? I remember reading about something like this a few years ago, but I can't remember where or when.

John M. Wargo (11826)
83 Hawthorne Ave., Akron, Ohio 44303

04821D

PROGRAM DESCRIPTION I

Program Title SIGNAL GENERATOR LEVELS

Contributor's Name Trevor H PHILLIPS

Address 30 Ngatitoa St.,

City POROIRUA State/Country N. Z. Zip Code

Program Description, Equations, Variables Program converts between the various calibration systems used on signal generators allowing a level to be converted to the units used on the instrument available. A user definable reference is also included to allow for instruments not covered by the program, or for use as a reference. The impedence of the generator is selectable and may be changed at any time during program use. All levels are stored for future reference.

$$P = \frac{V^2}{R} \qquad dB = 10 \log \frac{Power}{Ref.\ Power}$$

$$V = \sqrt{P \cdot R} \qquad P = \text{ANTILOG}\ \frac{dB}{10} \times Ref.\ Power$$

Necessary Accessories Nil

Operating Limits and Warnings Watch out for some signal generators which are calibrated in E.M.F. as opposed to P.D. a factor of 2 can result. (6 dB) Always enter the impedence before starting conversions.

Reference(s)

PROGRAM DESCRIPTION II

Sample Problem (Sketch if Desired) A signal of -100 dBm is required, The only available signal generator is calibrated in dBu, and has a 50 ohm output. Find the level to set it to.

EXAMPLE 2.

The sensitivity of a receiver for 12 dB sinad is specified as 0.5 uV ± 3 dB. Two sets were measured as having 0.7 uV. and 0.9 uV sensitivity respectively. The input impedence of the sets is 75 ohms. are they in spec?

EXAMPLE 3.

What voltage will be applied to a 100 ohm load by a 80 dBf signal?

SOLUTION:

Input	Function	Display		Comments
50	f a	50.0		Impedence.
100	CHS	100.	-15	-100 dBm. (power)
	C	6.99		6.99 dBu = -100 dBm.
	B			
EXAMPLE 2.				
75	f a	75.0		Impedence
0.5	EEX 6 CHS	.5	-06	spec. level in volts.
	A	3.33	-15	converted to power
	h SF 3			Set flag to load reference.
	f e	0.00		Reference loaded.
0.7	EEX 6 CHS	.7	-06	1st. sensitivity in volts.
	A	6.53	-15	converted to power
	f e	2.92		2.92 dB above ref (in spec).
0.9	EEx 6 CHS	.9	-06	2nd sensitivity in volts
	A	10.8	-15	converted to power
	f e	5.11		5.11 dB above ref. (out of spec.)
EXAMPLE 3.				
100	f a	100		Load in Ohms.
80	f d	100	-09	dB f to power
	A	3.16	-03	Conve t to voltage 3.16 mV

04821D
USER INSTRUCTIONS

STEP	INSTRUCTIONS	INPUT	FUNCTION	SIZE (HP-41C) DISPLAY
1	Load side 1 and 2 of card.			
2	Enter impedence of generator	Ohms	f a	Ohms
3	Enter signal level	Volts	A	Power
		dB u	B	"
		dB m	C	"
		dB 6m	D	"
		dB W	E	"
		dB 100u	f b	"
		dB 100m	f c	"
		dB f	fd	"
4	If this is the reference for a user defined level		h SF 3 f e	
4	For output in selected units press key as above			Units reqd.
5	For conversion to user reference		f e	dB user
5	For conversion to a different impedence; Enter new impedence	Ohms	f a	Ohms
	Press key for units required			Equivalent Power in units selected
6	To reviev any previously entered units ; Keys A to E are stored in registers A to E respectively. Keys f a to f e are stored in registers 1 to 5 sequentialy, ie f c (dB 100 mV) in R 3		RCL - - -	

PROGRAM LISTING

☒ 67 ☐ 97 ☐ 41C

STEP/ LINE	KEY ENTRY	KEY CODE (67/97 only)	COMMENTS
0 01	g LBL A	32 25 11	OHMS
	h P? 3	35 71 03	
	STO 1	33 01	
	RCL 1	34 01	
	h RTN	35 22	
	f LBL A	31 25 11	
	h P? 3	35 71 03	Find Volts.
	GTO 1	22 01	
	RCL 0	34 00	
10	RCL 1	34 01	
	X	71	
	f $\sqrt{x}$	31 54	
	STO A	33 11	
	h RTN	35 22	
	f LBL 1	31 25 01	
	STO A	33 11	Enter Volts.
	g x^2	32 54	
	RCL 1	34 01	
	÷	81	
20	STO 0	33 00	
	h RTN	35 22	
	f LBL B	31 25 12	
	h P? 3	35 71 03	Find dB uV.
	GTO 2	22 02	
	RCL 0	34 00	
	EEX	43	
	6	06	
	CHS	42	
	g x^2	32 54	
30	RCL 1	34 01	
	÷	81	
	f GSB 9	31 22 09	
	STO B	33 12	
	h RTN	35 22	
	f LBL 2	31 25 02	
	STO B	33 12	Enter dB uV.
	f GSB 0	31 22 00	
	EEX	43	
	6	06	
40	CHS	42	
	g x^2	32 54	
	RCL 1	34 01	
	÷	81	
	X	71	
	STO 0	33 00	
	h RTN	35 22	
	f LBL C	31 25 13	
	h P? 3	35 71 03	Find dB m.
	GTO 3	22 03	
50	RCL 0	34 00	

STEP/ LINE	KEY ENTRY	KEY CODE (67/97 only)	COMMENTS
51	EEX	43	
	3	03	
	CHS	42	
	f GSB 9	31 22 09	
	STO C	33 13	
	h RTN	35 22	
	f LBL 3	31 25 03	
	STO C	33 13	Enter dB m.
	f GSB 0	31 22 00	
60	EEX	43	
	3	03	
	CHS	42	
	X	71	
	STO 0	33 00	
	h RTN	35 22	
	f LBL D	31 25 14	
	h P? 3	35 71 03	Find dB 6 mW.
	GTO 4	22 04	
	RCL 0	34 00	
70	6	06	
	EEX	43	
	3	03	
	CHS	42	
	f GSB 9	31 22 09	
	STO D	33 14	
	h RTN	35 22	
	f LBL 4	31 25 04	
	STO D	33 14	Enter dB 6 mW.
	f GSB 0	31 22 00	
80	6	06	
	EEX	43	
	3	03	
	CHS	42	
	X	71	
	STO 0	33 00	
	h RTN	35 22	
	f LBL E	31 25 15	
	h P? 3	35 71 03	Find dB W.
	GTO 5	22 05	
90	RCL 0	34 00	
	f LOG	31 53	
	1	01	
	0	00	
	X	71	
	STO E	33 15	
	h RTN	35 22	
	f LBL 5	31 25 05	
	STO E	33 15	Enter dB W.
	1	01	
00	0	00	

Note: Refer to "HP-41C OWNER'S HANDBOOK AND PROGRAMMING GUIDE" for specific information on keystrokes. The Function Index is found at the very back of the Handbook. Refer to Appendix E in 67 or 97 "OWNER'S HANDBOOK AND PROGRAMMING GUIDE" for exact keystrokes.

PROGRAM LISTING

☒ 67 ☐ 97 ☐ 41C

STEP/ LINE	KEY ENTRY	KEY CODE (67/97 only)	COMMENTS
1 01	$\div\frac{}{x}$	81	
	g 10^x	32 53	
	STO 0	33 00	
	h RTN	35 22	
	g LBL b	32 25 12	- - - - - - - - - -
	h F? 3	35 71 03	Find dB 100 uV.
	GTO 6	22 06	
	RCL 0	34 00	
	EEX	43	
1 10	4	04	
	CHS	42	
	g x^2	32 54	
	RCL 1	34 01	
	$\div$	81	
	f GSB 9	31 22 09	
	STO 2	33 02	
	h RTN	35 22	- - - - - - - - - -
	f LBL 6	31 25 06	
	STO 2	33 02	Enter
1 20	f GSB 0	31 22 00	dB 100 uV.
	EEX	43	
	4	04	
	CHS	42	
	g x^2	32 54	
	RCL 1	34 01	
	$\div$	81	
	X	71	
	STO 0	33 00	
	h RTN	35 22	- - - - - - - - - -
1 30	g LBL c	32 25 13	
	h F? 3	35 71 03	Find dB 100 mV.
	GTO 7	22 07	
	RCL 0	34 00	
	.	83	
	1	01	
	g x^2	32 54	
	RCL 1	34 01	
	$\div$	81	
	f GSB 9	31 22 09	
1 40	STO 3	33 03	
	h RTN	35 22	
	f LBL 7	31 25 07	- - - - - - - - - -
	STO 3	33 03	Enter
	f GSB 0	31 22 00	dB 100 mV.
	.	83	
	1	01	
	g x^2	32 54	
	RCL 1	34 01	
	$\div$	81	
1 50	X	71	

STEP/ LINE	KEY ENTRY	KEY CODE (67/97 only)	COMMENTS
1 51	STO 0	33 00	
	h RTN	35 22	- - - - - - - - - -
	g LBL D	32 25 14	
	h F? 3	35 71 03	Find dB fW.
	GTO 8	22 08	
	f GSB C	31 22 13	
	1	01	
	2	02	
	0	00	
1 60	+	61	
	STO 4	33 04	
	h RTN	35 22	- - - - - - - - - -
	f LBL 8	31 25 08	
	STO 4	33 04	Enter dB fW.
	1	01	
	2	02	
	0	00	
	-	51	
	GTO 3	22 03	- - - - - - - - - -
1 70	g LBL e	32 25 15	
	h F? 3	35 71 03	USER
	STO 5	33 05	
	RCL 5	34 05	
	GTO 9	22 09	- - - - - - - - - -
	f LBL 9	31 25 09	
	$\div$	81	Sub-routine
	f LOG	31 53	
	1	01	
	0	00	
1 80	X	71	
	h RTN	35 22	- - - - - - - - - -
	f LBL 0	31 25 00	
	1	01	Sub-routine
	0	00	
	$\div\frac{}{x}$	81	
	g 10^x	32 53	
	h RTN	35 22	- - - - - - - - - -
90			
00			

Note: Refer to "HP-41C OWNER'S HANDBOOK AND PROGRAMMING GUIDE" for specific information on keystrokes. The Function Index is found at the very back of the Handbook. Refer to Appendix E in 67 or 97 "OWNER'S HANDBOOK AND PROGRAMMING GUIDE" for exact keystrokes.

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS

0 Last power
1 Impedence
2 dB 100 uV
3 dB 100 mV
4 dB fW.
5 User power

A Volts
B dB uV
C dB mW
D dB 6mW
E dB W.

STATUS

SIZE 187		TOT. REG.		USER MODE	
ENG	X	FIX	SCI	ON	OFF
DEG	X	RAD	GRAD		

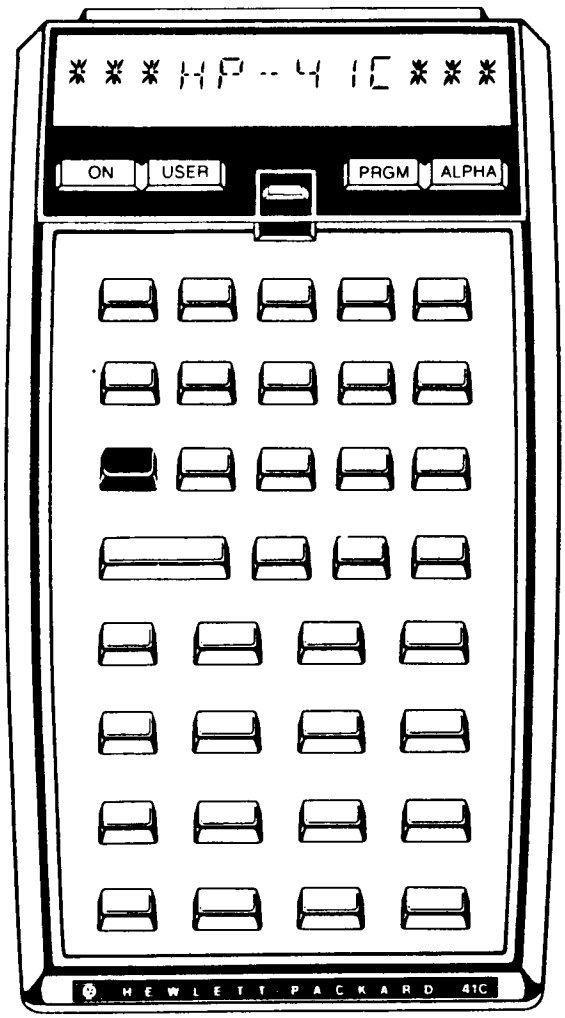
FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
3		Data entry	Units required

ASSIGNMENTS

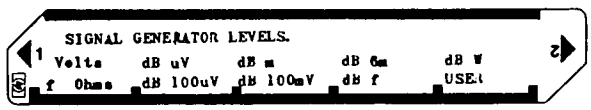
FUNCTION	KEY	FUNCTION	KEY
Volts	A	Ohms	a
dB uV	B	dB 100 uV	b
dB m	C	dB 100 mV	c
dB 6 mW	D	dB f	d
dB W	E	User	e

KEYBOARD CARD LABELING



KEYBOARD

CARD



R/S

SIGNAL GENERATOR LEVELS
HP-41 Translation

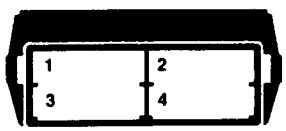
By Ross Cooling #12433
R.R. #1
Kimberley, Ontario
Canada NOC 1GO

This is a translation of a HP-67 program called Signal Generator Levels (number 4821 in the HP-67 library).

This program converts between the various calibration systems used on signal generators allowing a level to be converted to the units used on the instrument that's available. A user definable reference is also included to allow for instruments not covered by the program, or for use as a reference. The impedance of the generator is selectable and may be changed at any time during program use. All levels are stored for future reference. Watch out for some signal generators calibrated in E.M.F. as opposed to P.D., a factor of 2 (6 dB) can result. Always enter the impedance before starting conversions.

The minimum SIZE is 011. No printer is used. The top row of keys are used so any key assignments must be removed for this program. The keys are used as follows:
1) A is used for Volts
2) a is used to set impedance in ohms

SYSTEM CONFIGURATION



3) B is used for dB uV
4) b is used for dB 100uV
5) C is used for dB m
6) c is used for dB 100mV
7) D is used for dB 6m
8) d is used for dB f
9) E is used for dB W
10) e is used for USER reference. To set it first do SF 22 then press e.

Let's try some examples to make all of this clearer.

1. A signal of -100 dBm is required. The only available signal generator is calibrated in dBu, and has a 50 ohm output. Find the level to set to.

First, set the impedance level. Then set to the desired value and type. Finally convert to available type.
50, press a
-100, press C
press B to see 6.99 (dBu)

2. The sensitivity of a receiver for 12 dB sinad is specified as 0.5 uV ±3 dB. Two sets were measured as having 0.7 uV and 0.9 uV sensitivity respectively. The input impedance of the sets is 75 ohms. Are the two sets in spec?

Again, first set the impedance. Then set the specification level in volts. Use this value as the USER standard (remember to do SF 22 first). Enter the first voltage and convert to USER standard, then do the same for the second voltage.
75, press a
0.5 -06, press A, SF 22, press e
0.7 -06, press A
press e to see 2.92 (in spec. <3 dB)
0.9 -06, press A
press e to see 5.11 (out of spec. >3 dB)

3. What voltage will be applied to a 100 ohm load by a 80 dBf signal?

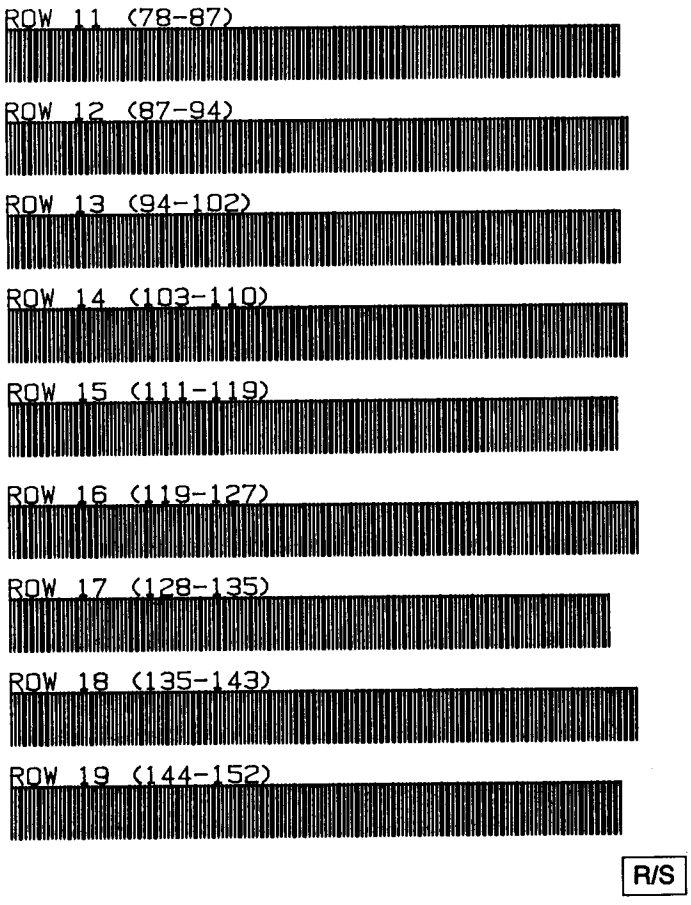
Again, start by setting the impedance. Enter the desired value and type, then convert.
100, press a
80, press d
press A to see 3.16 -03 (3.16 mV)

This should give you the idea of how to use this program. It's very powerful so just read over the instructions, try the examples then try your own values. I hope this program can help you with your conversions!

```
      PRP ""
01♦LBL "SIGNL"
  02♦LBL a
  03 FS?C 22
  04 STO 01
  05 RCL 01
  06 RTN

  07♦LBL A
  08 FS?C 22
  09 GTO 01
  10 RCL 00
  11 RCL 01
  12 /
  13 SQRT
  14 STO 06
  15 RTN

  16♦LBL 01
  17 STO 06
  18 X↑2
  19 RCL 01
  20 /
  21 STO 00
  22 RTN

  23♦LBL B
  24 FS?C 22
  25 GTO 02
  26 RCL 00
  27 E-12
  28 RCL 01
  29 /
  30 XEQ 09
  31 STO 07
  32 RTN

  33♦LBL 02
  34 STO 07
  35 XEQ 00
  36 E-12
  37 RCL 01
  38 /
  39 *
  40 STO 00
  41 RTN
```

```
42♦LBL 12
43♦LBL C
44 FS?C 22
45 GTO 03
46 RCL 00
47 E-3
48 XEQ 09
49 STO 08
50 RTN

51♦LBL 03
52 STO 08
53 XEQ 00
54 E-3
55 *
56 STO 00
57 RTN

58♦LBL D
59 FS?C 22
60 GTO 04
61 RCL 00
62 6 E-3
63 XEQ 09
64 STO 09
65 RTN

66♦LBL 04
67 STO 09
68 XEQ 00
69 6 E-3
70 *
71 STO 00
72 RTN

73♦LBL E
74 FS?C 22
75 GTO 05
76 RCL 00
77 LOG
78 10
79 *
80 STO 10
81 RTN

82♦LBL 05
83 STO 10
84 XEQ 00
85 STO 00
86 RTN

87♦LBL b
88 FS?C 22
89 GTO 06
90 RCL 00
91 E-8
92 RCL 01
93 /
94 XEQ 09
95 STO 02
96 RTN
97♦LBL 06
98 STO 02
99 XEQ 00
100 E-8
101 RCL 01
102 /
103 *
104 STO 00
105 RTN

106♦LBL c
107 FS?C 22
108 GTO 07
109 RCL 00
110 .01
111 RCL 01
112 /
113 XEQ 09
114 STO 03
115 RTN

116♦LBL 07
117 STO 03
118 XEQ 00
119 .01
120 RCL 01
121 /
122 *
123 STO 00
124 RTN
125♦LBL d
126 FS?C 22
127 GTO 08
128 XEQ 12
129 120
130 +
131 STO 04
132 RTN

133♦LBL 08
134 STO 04
135 120
136 -
137 GTO 03

138♦LBL e
139 FS?C 22
140 STO 05
141 RCL 05

142♦LBL 09
143 /
144 LOG
145 10
146 *
147 RTN

148♦LBL 00
149 10
150 /
151 10↑X
152 END
```

SIGNL
REGISTERS: 36
ROW 1 (1-3)
ROW 2 (4-13)
ROW 3 (14-24)
ROW 4 (25-31)
ROW 5 (32-39)
ROW 6 (40-47)
ROW 7 (48-54)
ROW 8 (55-62)
ROW 9 (62-69)
ROW 10 (69-77)



TWO MATH ROUTINES

By Philip T. Frohne #9660
PPC-MDC Chapter Coordinator
St. Louis, MO 63136-6102

CALCULATING CONGRUENCE

Calculating congruence is useful in finding prime numbers, factoring, and RSA (Rivest, Shamir, & Adelmen) public key cryptography. The technique presented here allows the manipulation of extremely large numbers in a format easily handled by the HP-41C and other programmable calculators.

The following equation is used:

$$X^{Enn} \equiv R \pmod{Z}$$

** "Enn" is the exponent in 41C notation **

X Enn is said to be congruent to R modulo {MOD} Z. R is the remainder after X is raised to the power of Enn and the result divided by the number Z, called the modulus of X Enn. The remainder after division of the number by the modulus is the residue of the number. Note: X and Enn must be positive integers.

The routine presented here works by using the theorem from number theory that states, "The residue of the square of a number is congruent to the square of the residue of the number".

The routine is executed by initializing the stack: "X":modulo, "Y":Enn, "Z":X and executing "CC" (X, ENTER, Enn, ENTER, mod, ENTER, XEQ "CC").

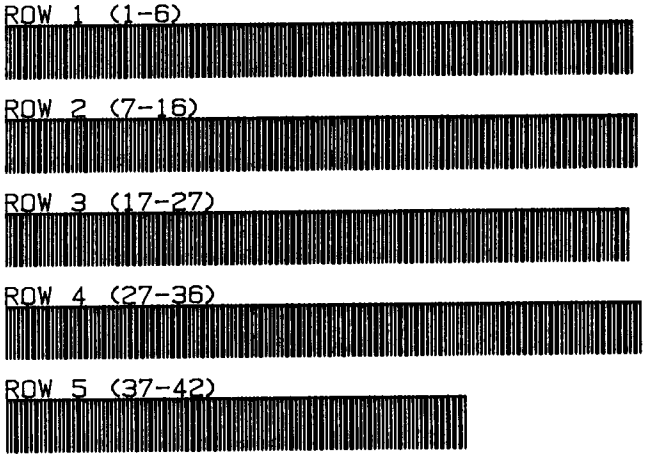
At the beginning of the routine, R equals 1. Enn is tested to be odd or even. If Enn is even, Enn is halved and X is squared mod Z. The new values are stored for X and Enn. If Enn is odd, R is changed to X times R mod Z. Enn is then tested to equal 1. If Enn does not equal 1, it is halved (integer portion used) and X is squared mod z. The test for Enn being even or odd is repeated as the routine loops until the Enn equals 1. When Enn equals 1, the routine ends.

Example:

$2^{43} \equiv R \pmod{5}$ calculates R to be equal to 3

```
01*LBL "CC"     15 RCL [        29 X↑2
02 STO [        16 MOD          30 RCL [
03  E           17 X<>Y         31 MOD
04 R↑           18 RCL \        32 X<>Y
05 R↑           19  E           33 GTO 00
06 STO \        20 X=Y?         34*LBL 02
07*LBL 00       21 GTO 03       35 RDN
08 2            22 RDN          36 RCL \
09 MOD          23 2            37 2
10 X=0?         24 /            38 /
11 GTO 02       25 INT          39 GTO 01
12 RDN          26*LBL 01       40*LBL 03
13 ST* Y        27 STO \        41 R↑
14 X<>Y         28 X^2          42 END
```

CC
REGISTERS: 9



CALCULATING LARGE FACTORIALS

Most calculators that have a factorial function can internally calculate factorials only up to N!=69. Beyond 69, approximations must be used because of processor limitations.

Routine "LFACT" uses the sharpened Stirling formula to approximate large factorials.

Place the factorial to be calculated in "X" and execute "LFACT". At completion, the

number in "X" is the approximated mantissa and the number in "Y" is the exponential power ("X" E"Y").

Example:

101! = 9.4259 E159

```
01♦LBL "LFACT"      17 *          33 X<>Y
02 ENTER↑           18 139        34 PI
03 STO Z            19 X<>Y       35 *
04 12               20 /          36 2
05 *                21 -          37 *
06 1/X              22  E         38 SQRT
07 X<>Y             23 +          39 LOG
08 X!2              24 LOG        40 +
09 288              25 X<>Y       41 ENTER↑
10 *                26  E         42 INT
11 1/X              27 E↑X        43 X<>Y
12 +                28 /          44 FRC
13 X<>Y             29 LOG        45 10↑X
14 3                30 R↓         46 END
15 Y↑X              31 *
16 51840            32 +
```

LFACT
REGISTERS: 10
ROW 1 (1-4)

ROW 2 (4-14)

ROW 3 (15-21)

ROW 4 (22-34)

ROW 5 (35-46)

ROW 6 (46-46)

R/S

Questions & Answers

By Ross Cooling #12433

QUESTION --

I'm a new member to PPC and I have some questions I hope you can answer.
1) Is there an annotated listing of the ZENROM?
2) Did Jim De Arras publish a second part to his HP-41 Bus article in PPCJ V7N3?
3) Is there a circuit board available for the design of the MLDL published in PPCJ V9N3? Or a finished connector/cable assembly to connect an MLDL with the HP-41?
4) With my ZENROM I found out how to apply and remove the private function from the catalog 1 mode. Is this old news?

Robert Reid (13240)
330 Colorado Avenue #3
Pueblo, CO
U.S.A. 81004-2036

ANSWER --

Welcome to PPC and we hope you enjoy it!
1) I wrote Zengrange, publishers of ZENROM, and they aren't releasing the annotated source. Of course with the ZENROM you can disassemble it, but you won't have any labels or comments.
2) Unfortunately I don't believe a second part was published.
3) I'm not aware of a source for either item. Most people probably would have wire-wrapped their board. Even ERAMCO on my MLDL just used an old module (probably from combining modules) and a standard flat 12 ribbon cable.
4) There are a number of ways people have found of removing PRIVATE, some of which have been published. The ZENROM certainly makes it easier, especially with RAMED. So probably it is old news.
(I have transferred the ZENROM MCODE to my computer and disassembled it there, but haven't annotated it yet. If that's of any help to you you can write me. I also found out that Lynn Wilkins made an MLDL II, but can't find any schematic. If anyone knows of one please write or call me. I'm also trying to come up with an improved version, with the help of others.)

QUESTION --

I need a program that creates pre-specified key assignments. I have modified an existing key assignment program to do this, but it's too long and too slow. I know it's possible to have it run faster, but don't know enough about synthetic programming to be able to do it. Here's what it does:

First, it clears existing key assignments.
Next, it makes ten calls to set the new assignments like this:

```
161  (XROM 06,02)
130
11   (Key 11)
XEQ 03 (Key assignment portion)
```

This assigns sequential XROM numbers to the top row of keys, and their shifts. ie.

```
XROM 06,03     06,05     06,07     etc.
XROM 06,02     06,04     06,06     etc.
```

Is there anything that exists that does this, or is there anyone who could help me? If there is I would greatly appreciate it. I don't think this is all that complex, but it's beyond me!

Warren Furlow (13223)
5595 Coronation Ct.
Dunwoody, GA
U.S.A. 30338

ANSWER --

Wow! Warren, your requested program certainly is *very* specific. You don't mention which HP-41 you have, but I get the impression it's not a HP-41CX so you don't have the PASN function. This request would be trivial with a CCD module using SAVEK, GETK, and MRGK to use extended memory to save key assignment combinations and switch or merge them very easily. There isn't any program I'm aware of to specifically assign keys as you want (not in the PPCJ at least). Does anyone else know of a program to do this? If you want to send me your existing program, specific assignment examples, your HP-41 model type, and a list of what modules you have I'll try my hand at trying to improve things for you!

In answer to some of Rick Wenger's questions:
(V14N4P35-36)

3) Yes, the HP-71 does support PEEK, POKE and CALL, but certain areas are not to be PEEKed or POKEd; with a FORTH ROM, nothing is sacred.

4) I doubt that Casio put a full assembler in the PC-5. I had one for a couple of days; I returned it when I found out that it was a simulator.

5) As far as I can tell, neither my TI-74 nor my TI-95 have machine language accessibility. The CALL keyword is used to call special subprograms, some of which are built-in. The subprograms can pass variables; of the programs provided is called KEY and works like the keyword INKEY$ found in some BASICs. You can also RUN programs in modules which fit in the one port in the TI-74. One of the programs is in the LRN PASCAL module, called PASCAL. This module, by the way, is a great way to learn some of the PASCAL language.

6) I would be delighted to provide reviews of the TI machines.

7) I believe that there is still a publication for TI machines.
Try writing to
TI PPC Notes
P.O. Box 1421
Largo, FL 34294

David Motto (2339)
2754 Granada, Apt. 2A
Jackson, MI 49202

KEY ASSIGNMENTS TO FILE

By Ritz Gerold #13025
3981 Bellwald
Switzerland

After reading in the book "Synthetic Programming Made Easy" by Keith Jarett about writing alarms to files or cards, I thought it would be nice to do the same thing with the key assignments. I wrote a program that saves the registers "├" and "e", too. But after my exams I finally had the time to read the back issues. In the journal V10N1 page 60 I found a program written by Tapani Tarvainen which was much better than mine, using the fact that GETP restores the registers "├" and "e". But I don't like that (after the modification in PPCCJ V10N2P3) the ID-bytes of the buffers are normalized. So I changed the program. Also my program doesn't go twice through the key assignment registers to get the needed filesize. I used the function RESZFL although the execution time of this function depends on the contens of the extendet memory. The comparison shows that my program is between two and five seconds faster.
As Tapani described in his article we need a programfile which is small and doesn't have any side effects. For this purpose we create a datafile of size one and change it into a programfile.
This is one of many ways you can do this:
1) remoove the X-memory modules if you have any (SX woun't work with them).
2) 1 in X-register; " " (space) in alpha XEQ"CRFLD".
3) CLA
4) XTOA (still 1 in X-register)
5) ASTO X
6) EMROOM
7) 69 +
8) XEQ"SX"
If you don't loose this dummy programfile called " ", you only have to do this procedure once.
To the program:
SAVEK : saves the key assignments to the file specified in alpha.
GETK : gets the key assignments from the file specified in alpha clearing the old key assignments.
MRGK : Adds the key assignments from the file specified in alpha to the old assignments (in case of an overlapping both assignments are lost).
SKX : "clears" the key assignments.
RKX : gets back the key assignments cleared by SKX.

```
01*LBL "SAVEK"
02 1
03 CRFLD
04 17E
05 "0"
06 RCL [
07 XROM "0M"
08 SIGN
09 SF 25
10*LBL 01
11 RDN
12 RCL IND Y
13 ""
14 X<> [
15 "├*"
16 X<> \
17 X≠Y?
18 GTO 02
19 ARCL c
20 X<> \
21 STO IND Z
22 SAVEX
23 CLX
24 1
25 ST+ T
26 ST+ Z
27 R↑
28 RESZFL
29 FC? 25
30 GTO 03
31 RDN
32 GTO 01
33*LBL 02
34 X<>Y
35 CLA
36 CLX
37 26
38 XTOA
39 X<> [
40 X=Y?
41 "
42 X≠Y?
43 " "
44 X<>Y
45 ARCL IND Z
46 X≠0?
47 X<> [
48 ST- IND Z
49*LBL 03
50 X<> L
51 X<> c
52 FC?C 25
53 GTO 04
54 R↑
55 1
56 -
57 RESZFL
58 RTN
59*LBL 04
60 "NOT ENOUGH ROOM"
61 AVIEW
62 RTN
63*LBL "GETK"
64 CLKEYS
65*LBL "MRGK"
66 CLX
67 SEEKPTA
68 FLSIZE
69 E
70 +
71 SIZE?
72 +
```

```
73 PSIZE
74 LASTX
75 PSIZE
76 FLSIZE
77 176
78 STO Z
79 +
80 XROM "E?"
81 RCL Y
82 -
83 16
84 -
85  E3
86 ST/ Z
87 X↑2
88 /
89 +
90 +
91 XROM "OM"
92 X<>Y
93 REGMOVE
94  E-3
95 -
96 GETRX
97 X<>Y
98 X<> c
99*LBL "RKX"
100 " "
101 GETP
102 RTN
103*LBL "SKX"
104 .
105 STO '
106 STO e
107 RDN
108 END
```

If you are shure that you have enough room in the main and the extended memory you may use the second version that doesn't check it amd therefore is shorter (221 or 179 bytes).

```
01*LBL "SAVEK"
02 I
03 CRFLD
04 176
05 "0"
06 RCL [
07 XROM "OM"
08 SIGN
09*LBL 01
10 RDN
11 RCL IND Y
12 ""
13 X<> [
14 "+*"
15 X<> \
16 X≠Y?
17 GTO 02
18 ARCL c
19 X<> \
20 STO IND Z
21 SAVEX
22 CLX
23 I
24 ST+ T
25 ST+ Z
26 R↑
27 RESZFL
28 RDN
29 GTO 01
30*LBL 02
31 X<>Y
32 CLA
33 CLX
34 26
35 XTOA
36 X<> [
37 X=Y?
38 "          "
39 X≠Y?
40 " "
41 X<>Y
42 ARCL IND Z
43 X≠0?
44 X<> [
45 STO IND Z
46 X<> L
47 X<> c
48 R↑
49 I
50 -
51 RESZFL
52 RTN
53*LBL "GETK"
54 CLKEYS
55*LBL "MRGK"
56 CLA
57 SEEKPTA
58 FLSIZE
59 176
60 STO Z
61 +
62 XROM "E?"
63 RCL Y
64 -
65 16
66 -
67  E3
68 ST/ Z
69 X↑2
70 /
71 +
72 +
73 XROM "OM"
74 X<>Y
75 REGMOVE
76  E-3
77 -
78 GETRX
79 X<>Y
80 X<> c
81*LBL "RKX"
82 " "
83 GETP
84 RTN
85*LBL "SKX"
86 .
87 STO '
88 STO e
89 RDN
90 END
```

Synthetic text lines:

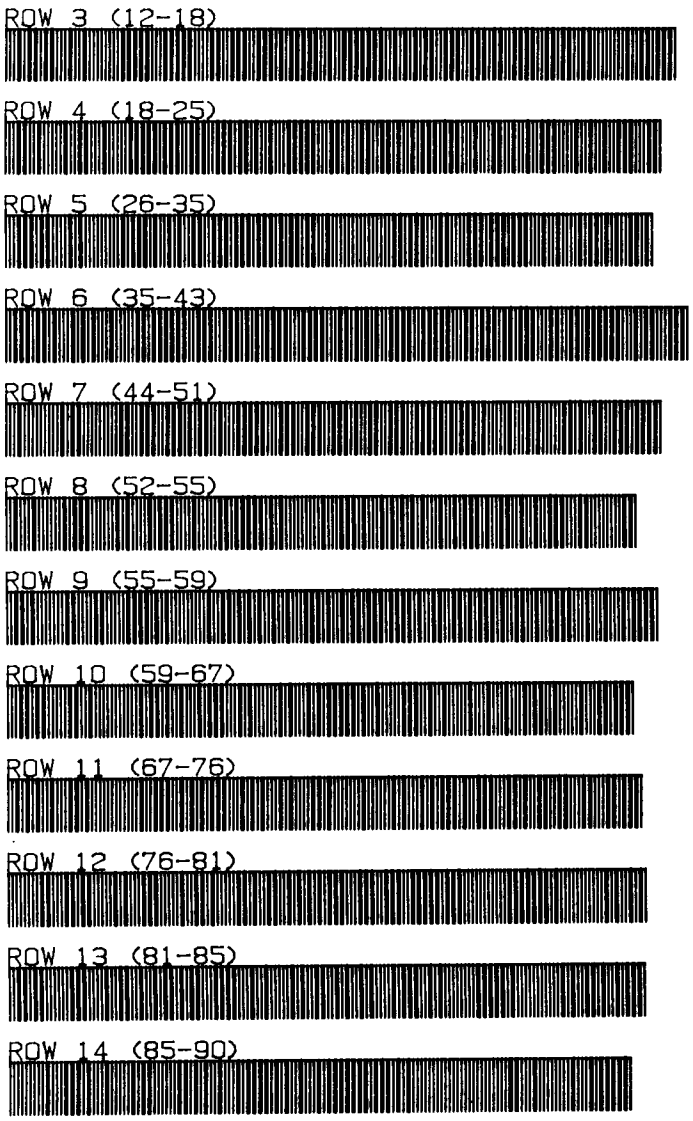
line number long version	line number short version	hex code
5	5	F110
13	12	F1F0
41	38	F1AA
43	40	F1BB

P.S. SKX and RKX have the same effect as SK and RK from the PPC ROM but instead of two data registers we need the program-file " " (space).

I wish you less MEMORY LOST's then I got while working with this stuff.

SAVEK
REGISTERS: 33
ROW 1 (1-4)
[redacted]
ROW 2 (4-11)
[redacted]
ROW 3 (12-18)
[redacted]
ROW 4 (18-25)
[redacted]
ROW 5 (26-33)
[redacted]
ROW 6 (34-42)
[redacted]
ROW 7 (43-50)
[redacted]
ROW 8 (51-59)
[redacted]
ROW 9 (60-60)
[redacted]
ROW 10 (60-63)
[redacted]
ROW 11 (64-67)
[redacted]
ROW 12 (68-76)
[redacted]
ROW 13 (76-83)
[redacted]
ROW 14 (83-92)
[redacted]
ROW 15 (93-99)
[redacted]
ROW 16 (99-103)
[redacted]
ROW 17 (103-108)
[redacted]
ROW 18 (108-108)
[redacted]

SAVEK
REGISTERS: 27
ROW 1 (1-4)
[redacted]
ROW 2 (4-11)
[redacted]



R/S

SWAP

EQUIPMENT WANTED !

Digital Cassette Drive HP 82161A
Thermal Printer IL HP 82162A
HP-IL Interface Module HP 82160A
Extended I/O Module HP 82183A
Plotter Module HP 82184A

EQUIPMENT FOR SALE !

Peripheral Printer HP 82143A
$175.00

Call: John J. Lally #13248
Spokane, Wa. (509) 448 8188
Evenings

WANTED: HP-55, HP-65, HP-70, HP-10 in working condition. Also, I need an HP-01 Watch in good condition with manual.Please contact Art Leyenberger, (201) 386-4254.

FOR SALE: HP-41C $85, HP82143A printer $135, AME Port Extender $75, PPC ROM $35, Box 10 data cassettes $35, Wickes' Synthetic Prog. $5; HP 41 Modules: Time - $25, X Functions - $25, Quad Memory - $25, X Memory - $25, Dual Memory - $12. Also, PPC Club Newsletters Vols 1 through 11#6 - $7/volume or $65 for entire set. Contact Art Leyenberger, (201) 386-4254.

PROGRAM EPHEM71

By D.M. Lazok #5454
P.O. Box 3105
El Segundo, CA 90245-8205

The program EPHEM71 calculates Sunrise-Sunset (SrSs) and/or Sun azimuth-elevation (?S) for any given time and date. This program has served me well in making antenna boresight determinations using the sun as a noise source.

Make sure you enter your correct Latitude (L0 in line 132), correct Longitude (L5 in line 138), correct standard Meridian (S0 in line 154) and correct Magnetic Variation [AKA magnetic deviation] (M7 in line 166). This insures fast and easy local calculations with a minimum of button pushing. Another convenience for those in the Americas, is the use of A,E,C,M,P (Atlantic ... Pacific) when the program prompts for a time zone. For those outside these five zones, enter an "X" and answer the next prompt with your nearest STANDARD TIME MERIDIAN (0,15,30...-30,-15 etc.). If <END LINE> is pressed at this prompt, the variable S0 will be filled with the value you specify in line 154.

Follow the prompts, and when asked for a start (or stop) DATE, notice the machine wants a MONTH and DAY separated by a comma. October 16 would be entered as: 10,16 <END LINE>. Leading zeros may be omitted. February 2nd may be entered as 2,2 <END LINE>. If 0,0 is entered for the **start date**, the machine will use the current date for calculations. TIME must always be entered. The format for **ENTERING TIME** is HH.hh, where 16.50 represents 16 Hours 30 minutes. 24 hour format is mandatory. Only the **TIME OUTPUT (display) is actual modulo 60 clock time** (HH.mm). As an example: Sunrise = 6.04 Sunset = 19.12 indicates literally, Sunrise = 06 Hours 04 minutes, Sunset = 19 Hours 12 minutes.

I generally enter a range of time, 6.0 to 7.0 (for example) with a 6 minute (.1 hr) increment. For finer grain, you may want to select an increment of 3 minutes (.05 hr). To leave the program, simply enter '3' or 'Q' at the Do: prompt.

Some obvious traps have been left out of the program. If you live on the North or South pole you will have to lie to the calculator. Also, I have reserved use of the numeral zero (0) as input for latitude, longitude, magnetic variation and date. This makes input for local calculations quick and easy.

Answers are fairly accurate, considering the liberal use of digit truncation techniques and a less than

elegant attempt to define a sphere "squashed at the poles" (or bulging at the equator).

So, the next time you are told to "get out of town by sun down", or want to tweak your E.M.E. array or need to know when hunting season really starts, set your HP-71B to the correct local time by tuning in the national bureau of standards, ask your local airport for the Lat/Lon and magnetic variation and you are all set!

If you don't feel up to entering the program yourself or would like a working copy of the program sent to you, please send a SASE with proper postage and 5 each HP-71 pull cards to my address as listed at the beginning of the program.

EPHEM71 is 4649 bytes long.

```
100 ! EPHEMERIS PGM FOR THE SUN
102 ! EXCLUSIVELY FOR THE HP-71B
104 !
106 ! D.M.LAZOK (5454)
108 ! POB 3105
110 ! EL SEGUNDO, CA. 90245-8205
112 ! (213)640-1273
114 !
116 ! VER:71.4.2
118 ! 09 MAY 87
120 !
122 DESTROY ALL @ FIX 1 @ RADIANS @ OPTION BASE 1
124 CFLAG -15 @ K1=15 @ K2=PI/2
126 !
128 BEEP 1400,.05
130 INPUT "Ur LAT DD.d (-)if S:";L0
132 IF L0=0 THEN L0=33.75 ! change L0=YOUR Latitude!
134 BEEP 1400,.05
136 INPUT "Ur LON DDD.d (-)if E:";L5
138 IF L5=0 THEN L5=118.45 ! change L5=YOUR Longitude!
140 BEEP 1400,.05
142 INPUT "TIME ZONE:A,E,C,M,P,X";Z$
144 IF Z$="A" THEN S0=60
146 IF Z$="E" THEN S0=75
148 IF Z$="C" THEN S0=90
150 IF Z$="M" THEN S0=105
152 IF Z$="P" THEN S0=120
154 IF Z$="" THEN S0=120
156 IF Z$="X" THEN INPUT "Ur STD TIME Meridian?";S0
158 BEEP 1400,.05
160 INPUT "Daylite Save (1=Y)? ";D
162 BEEP 1400,.05
164 INPUT "MagVar DD.d (-if E):";M7
166 IF M7=0 THEN M7=14.25 ! change M7=YOUR Mag Var!
168 BEEP 1400,.05
170 INPUT "Leap Yr (1=Y)?:";Q0
172 IF Q0=1 THEN D5=366 @ Q1=1 ELSE D5=365.24232 @ Q1=
0
174 K=360/D5
176 BEEP 1400,.05
178 INPUT "Do: 1)?S 2)SrSs 3)Q:";N1
180 IF N1)3 THEN 178
182 IF N1=3 OR N1=0 THEN 480
184 BEEP 1400,.05
186 INPUT "Start Date (MM,DD):";M1,D1
188 IF M1 OR D1)0 THEN GOSUB 452 ELSE N9=FP(DATE*.001)
*1000 @ N7=N9 @ GOTO 200
190 N9=N8
192 BEEP 1400,.05
194 INPUT "Stop Date (MM,DD)";M1,D1
196 IF M1=0 OR D1=0 THEN N7=0 @ GOTO 208 ELSE GOSUB 45
2
198 N7=N8
200 N5=N7-N9
202 IF N9)N7 AND N7()0 THEN DISP "FIRST ) LAST: RE-DO"
 @ N=2 @ GOTO 186
204 L1=RAD(L0)
206 IF FLAG(1) THEN L5=L6 @ CFLAG 1
208 IF N1=1 THEN L6=L5 @ SFLAG 1 @ L5=L5-1.5
210 IF L5)=0 THEN M0=L5-S0 ELSE M0=S0-L5
212 M9=M0/K1 @ M8=RAD(M0)
214 IF N1=2 THEN SFLAG 2 @ GOTO 310
216 BEEP 1400,.05
218 INPUT "START TIME(0-24):";N3 @ IF N3)24 THEN BEEP
950,.1 @ DISP "TIME TOO LARGE" @ GOTO 214
220 N3=N3-D
222 BEEP 1400,.05
224 INPUT "STOP TIME(0-24):";N4 @ IF N4)24 THEN BEEP 9
50,.1 @ DISP "TIME TOO LARGE" @ GOTO 224
226 N4=N4-D
228 IF N3)N4 AND N4()0 THEN DISP "START ) STOP--DO AGA
IN" @ N4=0 @ GOTO 214
230 BEEP 1400,.05
232 INPUT "INCREMENT (HH.h)?";I @ IF I=0 THEN I=1
234 B0=K2-L1
236 FOR N=N9 TO N7
238 GOSUB 424
240 GOSUB 400
242 FOR S=N3 TO N4 STEP I
244 S1=IP(S) @ S2=FP(S)*.6 @ S3=S1+S2
246 A1=K2-Z
248 IF S(12 THEN C=RAD((12-S)*K1)+E0+M8 ELSE C=RAD((12
-S)*K1)+E0+M8
250 E=(B0-A1)/2 @ F=(B0+A1)/2 @ G=C/2
252 X=COS(E)/(COS(F)*TAN(G))
254 Y=SIN(E)/(SIN(F)*TAN(G))
256 X1=ATN(X)*2
258 Y1=ATN(Y)*2
260 B=(X1+Y1)/2
262 A=X1-B
264 L=(B+A)/2 @ M=(B-A)/2
266 Z1=TAN(E)*SIN(L)/SIN(M)
268 C1=2*ATN(Z1)
270 A2=90-(DEG(C1)+.5)
272 IF C1(0 THEN A2=180-A2
274 A3=IP(DEG(A)+.5)
276 IF C1(0 AND A(0 THEN A3=180+A3
278 IF S)12 AND A3(180 AND A3)=0 THEN A3=A3+180
280 IF A3(0 THEN A3=A3+360
282 IF L1-Z)0 THEN 290
284 IF S=12 THEN A3=IP(360-M0-E1+.5) @ A2=90*COS(M8+E0
)-DEG(Z-L1)+.5 @ GOTO 286 ELSE 292
286 IF A3)=360 THEN A3=A3-360
288 GOTO 292
290 IF S=12 THEN A3=IP(180-M0-E1+.5) @ A2=90*COS(M8+E0
)-DEG(L1-Z)+.5
292 DISP "Sun is:" @ WAIT 1
294 FIX 2 @ DISP S3+D;"Local Time" @ BEEP @ WAIT 2
296 STD @ DISP A3;" True Azimuth" @ BEEP @ WAIT 2
298 IF A2(.1 AND A2)-.09 THEN A2=0
300 FIX 1 @ DISP A2;" Elevation" @ BEEP @ WAIT 2
302 BEEP 950,.1 @ INPUT "RE-DISPLAY? ";A$
304 IF A$="Y" THEN 294
306 NEXT S @ NEXT N
308 GOTO 178
310 ! D.M. LAZOK  PPC# (5454)
312 BEEP 1400,.05
314 INPUT "INCREMENT (IN DAYS):";X5
316 IF X5=0 OR X5)N5 THEN X5=1
318 FOR N=N9 TO N7 STEP X5
320 GOSUB 400
322 C1=K2
324 IF L1-Z(0 THEN T=IP(360-E1+M7-M0+.5) ELSE T=IP(180
-E1+M7-M0+.5)
326 IF T)=360 THEN T=T-360
328 A1=C1-Z
330 B0=C1-L1
332 S4=(A1+B0+C1)/2
334 ON ERROR GOTO 392
336 T1=SQR(SIN(S4-A1)*SIN(S4-B0)*SIN(S4-C1)/SIN(S4))
338 C5=T1/SIN(S4-C1)
```

```
340 C=2*DEG(ATN(C5))
342 A3=2*DEG(ATN(T1/SIN(S4-A1)))
344 C2=C/K1
346 S5=12-.075+E2+M9-C2
348 U1=IP(S5)
350 M2=(S5-U1)*60
352 M2=IP(M2+.5)/100
354 IF M2=.6 THEN M2=0 @ U1=U1+1
356 S5=U1+M2
358 S6=12+.057+E2+M9+C2
360 D2=IP(S6)
362 M3=(S6-D2)*60
364 M3=IP(M3+.5)/100
366 IF M3=.6 THEN M3=0 @ D2=D2+1
368 S6=D2+M3
370 IF L1-Z(0 THEN A2=90*COS(M8+E0)-DEG(Z-L1)+.5 ELSE
A2=90*COS(M8+E0)-DEG(L1-Z)+.5
372 GOSUB 424
374 FIX 2 @ BEEP @ DISP "SunUp=";S5+D;"@"; @ STD @ DIS
P IP(A3+.5);"" @ WAIT 3
376 BEEP @ FIX 2 @ DISP "SunSet=:";S6+D;"@"; @ STD @ D
ISP IP(360-A3+.5);"" @ WAIT 3
378 BEEP @ DISP "Zen:";T;"Az"; @ DISP IP(A2);"E1" @
WAIT 3
380 BEEP @ INPUT "RE-DISPLAY?";A$ @ IF A$="Y" THEN 374
382 NEXT N
384 CFLAG 2
386 GOTO 178
388 !
390 ! ERROR TRAP ROUTINE NEXT 4 LINES
392 OFF ERROR
394 DISP N;"/CANT DETERMINE"
396 GOTO 396
398 !
400 L9=RAD(279.575+K*N)
402 G9=RAD(356.967+K*N)
404 L2=L9+RAD(1.916*SIN(G9)+.02*SIN(2*G9))
406 D3=.39782*SIN(L2)
408 Z=ATN(D3/SQR(-(D3*D3)+1))
410 E6=-(104.7*SIN(L9))+596.2*SIN(2*L9)+4.3*SIN(3*L9)-
12.7*SIN(4*L9)
412 E5=E6-429.3*COS(L9)-2*COS(2*L9)+19.3*COS(3*L9)
414 E2=-(E5/3600)
416 E1=E2*K1
418 E0=RAD(E1)
420 RETURN
422 !
424 STD @ IF N(=31 THEN DISP "JAN ";N;" "; @ BEEP
426 IF N)31 AND N(=59+Q1 THEN DISP "FEB ";N-31;" "; @
BEEP
428 IF N)59+Q1 AND N(=90+Q1 THEN DISP "MAR ";N-(59+Q1)
;" "; @ BEEP
430 IF N)90+Q1 AND N(=120+Q1 THEN DISP "APR ";N-(90+Q1
);" "; @ BEEP
432 IF N)120+Q1 AND N(=151+Q1 THEN DISP "MAY ";N-(120+
Q1);" "; @ BEEP
434 IF N)151+Q1 AND N(=181+Q1 THEN DISP "JUN ";N-(151+
Q1);" "; @ BEEP
436 IF N)181+Q1 AND N(=212+Q1 THEN DISP "JUL ";N-(181+
Q1);" "; @ BEEP
438 IF N)212+Q1 AND N(=243+Q1 THEN DISP "AUG ";N-(212+
Q1);" "; @ BEEP
440 IF N)243+Q1 AND N(=273+Q1 THEN DISP "SEP ";N-(243+
Q1);" "; @ BEEP
442 IF N)273+Q1 AND N(=304+Q1 THEN DISP "OCT ";N-(273+
Q1);" "; @ BEEP
444 IF N)304+Q1 AND N(=334+Q1 THEN DISP "NOV ";N-(304+
Q1);" "; @ BEEP
446 IF N)334+Q1 THEN DISP "DEC ";N-(334+Q1);" "; @ BEE
P
448 FIX 1 @ RETURN
450 !
452 ON M1 GOTO 454,456,458,460,462,464,466,468,470,472
,474,476
454 N8=D1 @ RETURN
456 N8=31+D1 @ RETURN
458 N8=59+D1+Q1 @ RETURN
460 N8=90+D1+Q1 @ RETURN
462 N8=120+D1+Q1 @ RETURN
464 N8=151+D1+Q1 @ RETURN
466 N8=181+D1+Q1 @ RETURN
468 N8=212+D1+Q1 @ RETURN
470 N8=243+D1+Q1 @ RETURN
472 N8=273+D1+Q1 @ RETURN
474 N8=304+D1+Q1 @ RETURN
476 N8=334+D1+Q1 @ RETURN
478 !
480 BEEP 1400,.05 @ INPUT "Restart? ";A$ @ IF A$="Y" T
HEN 122
482 DEGREES @ FIX 2 @ CFLAG 1,2 @ BEEP @ END
```

R/S

HOW TO CONVERT HP-67/HP-97 PROGRAMS TO HP-41 FORMAT

By Ross Cooling #12433
R.R. #1
Kimberley, Ontario
Canada NOC 1GO

When the HP-41 was brought out in 1979 Hewlett-Packard was well aware of the large number of HP-67s and HP-97s in the marketplace. Since they had card readers built-in there would be a lot of programs on cards. To maintain compatibility to the HP-41, which is basically a superset of the HP-67s, the HP-41 card reader (82104A) was given the ability to translate about 95% of the HP-67 code to HP-41 code. The remaining 5% was handled by MCODE routines built into the HP-41 card reader (they all start with 7, except PRREG). This was excellent, but meant for some programs that the card reader was needed to run the program!

This article is written for those who want to be able to do conversions of HP-67/HP-97 programs to HP-41 format, but not require the card reader afterwards. You may find it interesting to read the article in V14N1P19 of the PPC Journal, and you should read chapter 3 of the HP-41 card reader manual.

The first step is to read in the HP-67/HP-97 card into the HP-41, but you must prepare for it first. You have to set the SIZE to a minimum of 026, be in USER mode, and be at the end of memory. You should be warned that the mode (DEG/RAD/GRAD, FIX/SCI/ENG, and number of decimal places) will change to what values the HP-67 had when the program was saved. The card(s) read in the same as for the HP-41, but will go into a PACKING step after loading.

Now that you have the program loaded remember for later the modes the HP-41 is now in as you may want to modify the program later to put these values into it. When you go into program mode you'll see the first program line is 01 LBL 67 (whether the card came from a HP-67 or HP-97). You can delete it and insert your own label, and then go to the bottom and put an END on if you like and delete the then unnecessary STOP. At this point in time you may want to get a listing of the program, to make editing easier, and run the program as is

to make sure it does work before making more modifications!

Now, let's talk about changing some of the functions starting with 7 to HP-41 functions. Naturally we'll start with the easy ones from pages 68 and 69 of the HP-41 card reader manual.

7DSP0 through 7DSP9 change the number of digits after the decimal point without concerning itself with the mode (FIX/SCI/ENG). The HP-41, as you well know changes both the mode and the decimal value at the same time. There is no way to just change the digits without getting into synthetic programming, and probably messing up the stack. The practical solution is to put in the current mode and the decimal value (ie. 7DSP3 to FIX 3, unless you know it should be SCI 3 or ENG 3). The 7FIX, 7SCI, and 7ENG operate the same way in that the mode changes regardless of the number of digits after the decimal point. Again, change it as appropriate to the program as a proper HP-41 instruction (ie. 7FIX to FIX 2).

7PRTX will print the X register if there's a printer, else it displays X. This can be changed to:

```
FS? 21, PRTX, FC? 21, PSE
```

Before we translate any more functions this would be a good time to discuss registers on the HP-67/HP-97. They have 26 registers in total split into three groups - a group of ten primary registers (Reg. 00-09 on the HP-41), a group of ten secondary registers (Reg. 10-19 on the HP-41), and six more primary registers A, B, C, D, E, and I (Reg. 20-25 on the HP-41). The primary registers are very similar to those on the HP-41 so you can store and recall directly from them. The secondary registers are sort of protected so you have to get at them indirectly (similar to registers over 99 on the HP-41). The two banks of ten registers can be swapped to enable direct access to what are now the primary ten. The register I is for indirect access, whereas any register on the HP-41 can be used for indirection.

Now that we understand how the HP-67/HP-97 registers are set up and how they're mapped onto the HP-41 registers let's translate some more functions.

7P<>S exchanges the primary and secondary registers which are 00-09 and 10-19. If you have an HP-41CX you can do:

```
0.010010, REGSWAP, RDN
```

(if the value in the T register is never used). For those without REGSWAP a brute force method is: X<> 00, X<> 10, X<> 00, ..., X<> 09, X<> 19, X<> 09. A more practical method I've found requires following what the program is doing. You'll probably find the 7P<>S functions to be in pairs with some instructions in between. If you're careful you should be able to get rid of the 7P<>S and change the registers from 00-09 to the corresponding values between 10 and 19. Try this out and run your program to verify it still works!

The function 7CLREG clears all the primary registers (00-09, 20-25), but not the secondary registers (10-19). If you have the CLRGX instruction on your HP-41 and the program doesn't need the T register you can use:

```
0.009, CLRGX, RDN, 20.025, CLRGX, RDN
```

A brute force method is :

```
X<> 00, CLX, X<> 00
```

substituting 01-09, and 20-25 for 00. You may notice in the program the steps:

```
7CLREG, 7P<>S, 7CLREG
```

which mean they're also clearing the secondary registers so you can just use:

```
0.025, CLRGX, RDN
```

or if you don't care about any other registers above 25 just use: CLRG.

For 7RCLΣ you can use: RCL 11, RCL 13 if you have your statistical registers starting at the default of register 11, otherwise adjust the register numbers as necessary.

The function 7PRSTK will print the contents of the stack (X,Y,Z,T) if a printer is attached, otherwise it will just display them. If you don't care about the display portion just use: FS? 21, PRSTK. If you do want the display portion it's more complicated and will require something like:

```
FS? 21
PRSTK
FS? 21
GTO 01
VIEW T
PSE
VIEW Z
PSE
VIEW Y
PSE
CLD
PSE
LBL 01
```

PRREG operates like 7PRSTK, but works with the primary registers 00-09 and 20-25. Again, if you don't care about the display and the T register isn't used then use:

```
FS? 21
GTO 01
0.009
PRREGX
RDN
20.025
PRREGX
RDN
LBL 01
```

If you do care about the display then use the above routine but just in front of the LBL 01 put in VIEW nn and PSE as necessary.

7GTOI is simply GTO IND 25 if the value in register 25 is between 0 and 19. If the value is negative, on the HP-67 it would cause the program counter to go backwards that number of steps! Clearly that's not possible with the HP-41 so even the card reader just says NONEXISTENT. 7GSBI is very similar to 7GTOI and can be replaced with XEQ IND 25. Again, negative values will produce NONEXISTENT.

The next four functions are probably the hardest to translate. They are 7DSZ, 7DSZI, 7ISZ, and 7ISZI. 7DSZ decrements the I register (Reg. 25 on the HP-41) and skips the

next instruction when I equals zero. 7ISZ does the same, except it increments instead of decrements. 7DSZI decrements the register pointed to by I and skips the next instruction if the register was zero (an indirect operation). 7ISZI does the same as 7DSZI, but decrements instead. If it was just the increment/decrement operation that was important, and the T register wasn't used, you could use:

```
1, ST- 25, RDN       for 7DSZ
1, ST- IND 25, RDN   for 7DSZI
1, ST+ 25, RDN       for 7ISZ
1, ST+ IND 25, RDN   for 7ISZI
```

You can also try using ISG/DSE, but you'll need to understand the program pretty well and make other changes probably. Sorry, but this is the best I can offer for these functions.

That covers all the functions provided by the HP-41 card reader. Some aren't too difficult to translate, and others are very difficult or impossible to do exactly as the HP-67 would. The important thing is to use some common sense when translating them and probably compromise somewhat (like the display aspects of the print functions).
Let's get back to the programs from HP-67s as translated by the HP-41 card reader. You'll also notice that the leading 1 is dropped from exponents (ie. 1 E2 is just E2). This is perfectly okay, and is actually how synthetic programming on the HP-41 started! You may also notice negative numbers have trailing negative signs, but this is okay too. LBLs A-E and a-e will be preceded by a LBL 10-14 and 15-19 respectively. These are there in case there is a GTO or XEQ of that routine to speed it up. You can safely remove all unnecessary LBLs.

Most other changes you can make aren't needed, but will tend to reduce the byte count and/or speed up the execution. They include compacting the registers (ie. if only registers 2, 10, and 25 are used you could change them to 1, 2, and 3), and making use of functions the HP-41 has which weren't on the HP-67 (ie. the instructions 2, * execute faster as ST+ X). Just be careful with all your changes and do a lot of testing, especially boundary cases.

Remember to include in your program, or documentation, the modes that the HP-41 needs to be in that you saved from the HP-67! Finally, rather than provide any examples here check PPC Journals starting with V14N4 (April 1987) to see HP-67 programs and my translated versions. For the most part they are translated to remain as close to their original, rather than try to improve them.

Lastly for all those who don't have HP-67s and don't think this article applies to them, remember that PPC has the HP-67/HP-97 program library with thousands of programs in it. These are available to you for a few dollars per program, and a little effort in translating them on your part (but now you know how!). Happy translating!

R/S

CORRESPONDENCE

PPC Club
P.O. Box 90579
Long Beach, CA 90809

Dear PPC:

Until recently, we were able to make an occasional exception to our usual policy of not supporting Hewlett-Packard calculator products that had been modified by other than authorized Hewlett-Packard service personnel.

However, due to recent changes in our repair procedures, all Hewlett-Packard calculator products received for service that have been modified will be returned unrepaired. This pertains to all non-HP modifications such as HP-41 speed-up kits, 28C memory expansion, etc.

Effective immediately, exceptions to this policy will no longer be made and all modified units will be returned to the customer unrepaired. Full details of our service and warranty policies are provided in the Owner's Handbook of calculator model.

Lori W. Fuszek
Service Administrative Supervisor
Hewlett Packard
Portable Computer Division
Corvallis, OR

Every arbitrary program is longer, less efficient and more expensive.

I think this is my third renewal — possibly my last. I'm still greatly underwhelmed. My referral to PPC was through the excellent book about the HP-71B by Joseph Horn. I soon found that PPC was little help for a 71 user. For every reference to the 71 there are 832 references to the HP-41. But I don't have a 41.

I learned RPN on the HP-97 while I was still working. Now I'm retired and have acquired an HP-15C and the fascinating HP-71B.

I still eagerly devour anything that concerns the 71, but find it exceedingly sparse. The books by Horn and Richard Harvey are mighty helpful. A non-HP book on programming the 15C helped there. Now I find another problem. I recently bought both the 18C and the 28C. I find there is a plethora of books on the 18C, but only the non-understandable HP manual for the 28C.

A few articles on programming the 28C are sorely needed in the PPC Journal. I've learned how to write a program for finding the area of a pizza, and also one for finding the volume of a water or gasoline tank, but duplicating what I considered a simple program on the 15C for updating multiple bank account balances is so obscure that it is beyond me. I hesitate to question HP, since my only question directed that way about the 71 was answered politely and warmly with an effective "Read the Manual." That's why I wrote the letter. I couldn't understand the Manual.

I appreciate that you have had your problems at PPC but I still think that it should do us guys who don't have 41's some good too.

Harold Hoots #12971
4666 Powers Boulevard
Decatur, IL 62521

Good point. This is your Journal, our material comes from you, our members. No input to PPC then no 71 output in the Journal. The cry from the members is more "Simple and Short Routines." There must be some 71 (and 41) programs out in the PPC world. Please send them in, we are anxious. Thanks.