TRAVERSE, INVERSE and SIDESHOTS
A POINT STORAGE CONVERSION
for the HP-41CV/CX SURVEYING PAC

Copyright © 1987 by Ted J. Kerber

All rights reserved. No part of this work covered by the copyright hereon may be reproduced or used in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without written permission of the author.

published by D'Zign **Land Survey & Development**
Pacifica, California 1987

ISBN 0-9616846-4-X

Introduction

Someone once said, "Find a need, and fill it" .... That, in essence, is what these booklets are all about. To fill the need for software in the field.

Every new book we have published has prompted response that the user wishes that it had ALSO included a program for .... Or, "I only need the programs for spirals, I haven't tried the rest of them yet."

Most people will buy one of our books for a particular program that it contains, not because they need all of the programs in that particular book. Those comments prompted this series of "do it yourself" booklets. You only buy the ones you have a need for.

This book contains editing procedures for turning your HP Surveying Pac into a point storage system, and instructions for modifying the **TRAVERSE, INVERSE AND SIDESHOT** program of the Surveying Pac into a system that can store 100+ coordinate pairs in memory.

Coordinates may be recalled by point number after a traverse has been run, to do further calculations, and the results of those calculations are also stored.

You need a 41, a survey pac and a little time for editing the programs. Extended memory is NOT needed for point storage.

An added feature of the Traverse modification is an AUTOMATIC COMPASS CORRECTION subroutine.

This book has been published in two versions, one with and one without magnetic cards enclosed. If you have the version with the cards, you will still find the editing lists helpful, and you can further modify the programs to suit your particular needs.

Utility Programs

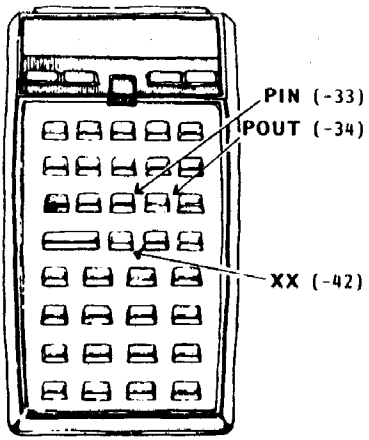
The Traverse, Inverse and Sideshots program contained in your HP Surveying Pac will respond in exactly the same way you are accustomed to, with all of the prompts and responses exactly as shown in the instruction manual you received with your module. At the same time, it will be storing the coordinate values for later use.

An automatic Compass Rule Adjustment program has been included in the modifications. When used, it replaces the original coordinate pairs with the adjusted pairs, and **automatically** outputs the adjusted bearings and distances of the courses.

The unadjusted data may be written to a data card, stored in extended memory or external memory before adjusting if you like.

There are three short UTILITY programs which must be in program memory as part of the operating system. Two are used for point number manipulation and one is used for automatically inversing between any stored coordinate pairs. A really fast layout routine with pre-stored coordinates.

"Point **IN**" (PIN) allows direct storage of point-numbered coordinates. "Point **OUT**" (POUT) will recall any coordinate pair whose point number is in the x-register when executed.



For convenience, assign these utility programs to the keyboard. They are assigned to the **shifted** functions of the button, as shown to the left.

These programs may be keyed in at any time, and should be left in program memory when ANY of the other routines are being used. The program for radial inversing is called "**XX**".

Let's begin by typing in the utility programs. Before beginning, execute a **GTO··** and enter **program mode** by using the following keystrokes:

Your display should show **00 REG** and a number, **NNN**. This last number is the number of program registers left in memory. For this program you will need 21 registers available. Type in the program steps shown in the listing below.*

```
01♦LBL "XX"
02 XROM "*IN"
03 XEQ "C*"
04 CF 10
05 XEQ "POUT"
06 XROM "NE"
07 AVIEW
08 ADV
09 STO 08
10 RDN
11 STO 07
12♦LBL 03
13 CF 01
14 "TO?"
15 PROMPT
16 XEQ "POUT"
17 DSE 13
18 STO X
19 XROM "INVERSE"
20 GTO 03
21 RTN
22♦LBL "POUT"
23 STO 13
24♦LBL "OUT"
25 RCL 13
26 2
27 *
28 19
29 +
30 ENTER↑
31 ENTER↑
32 1
33 -
34 RCL IND X
35 RCL IND Z
36 RTN
37♦LBL "PIN"
38 SF 10
39 STO 13
40 XROM "NE"
41♦LBL "IN"
42 RCL 13
43 2
44 *
45 19
46 +
47 X<>Y
48 STO IND Y
49 RDN
50 1
51 -
52 X<>Y
53 STO IND Y
54 RTN
55♦LBL "C*"
56 .019
57 0
58♦LBL 11
59 STO IND Y
60 ISG Y
61 GTO 11
62 RDN
63 RDN
64 RTN
65 END
```

The steps, XROM "NE" and XROM "INVERSE", are put in as XEQ "NE" and XEQ "INVERSE". The calculator will change the XEQ to XROM for you. For instance, the keystrokes for input of program step 06 are

Any portion of a program step which is included in quotation marks (" ") indicates that it is **alpha** input, and must be input with the calculator in **alpha mode**.

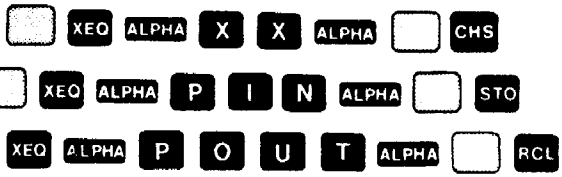
*If your edition of this book includes program cards, see Appendix A.

User instructions

The only program steps which might give you trouble are the "indirect" store and recall instructions. An example of the input for these is program step 34, which is input by keystroking

The only other one that may be misleading is step 49, RDN, this is the printout for "rolldown", and is shown on the keyboard as **R↓**. The ↓ symbol at each label is inserted by the calculator, and you don't need to input it.

Next, let's assign these programs to the keyboard so that we can try them out. Keystroke



Size your calculator to **040**, and we'll input some coordinates by point number, using **"PIN"** (assigned to the shifted **CHS** key, #-42). The keystroke instructions are as follows:

1 Input the point number, and stroke

2 **NX=?**

Input the north coordinate for the point, and stroke

3 **EX=?**

Input the east coordinate for the point, and stroke

That's all there is to it! Continue in the same manner until all of your coordinates have been stored. We'll do some keystroke examples on the next page, inputting a few coordinate pairs to practice with.

"**XX**" uses both of the shorter programs as sub-routines, and taps the Surveying Pac to do the inversing between the points. In the keystroke procedures which follow, it is assumed that "**XX**" has already been assigned to key #-41, the shifted CHS key. Just follow these simple steps:

1 Put the calculator into USER mode, (normal form is FIX 4, and in Degree Mode)

2 Input the beginning point number, and stroke [ ] CHS

output: **Nx=XXXXX.xxxx**

3 If a printer is attached the output is automatic. If no printer is present keystroke R/S

output: **Ex=XXXXX.xxxx**

R/S if no printer

TO?

4 Input the point number of the point you are inversing to, and stroke R/S

output: **AZ=DDD.mmss***

5 Stroke R/S

output: **HD=XXX.xxxx**

6** Stroke** R/S

output: **Ny=XXXXX.xxxx**

7** Stroke** R/S

output: **Ey=XXXXX.xxxx**

*Bearing output will be shown instead of azimuth when **FLAG 00** is set.

These last two steps are automatic with the printer attached. If no printer is present, continue stroking R/S until the prompt **TO? appears.

Keystroke Examples

Continue calculating RADIAL ties by inputting the next point number each time the **TO?** prompt appears, and stroking

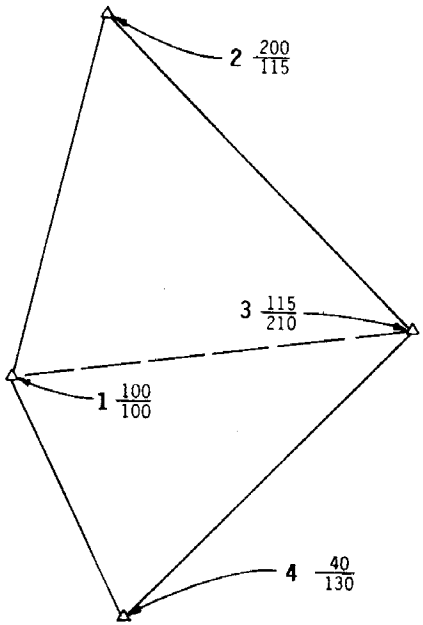
R/S

You may also use this routine to do an **inverse traverse** between pre-stored points. The program will "move up" to the new point when **FLAG 01** is set.

For INVERSE TRAVERSE: Each time the **TO?** prompt appears, input the next point number and stroke

[ ] 7 0 1 before stroking R/S

For examples of how this program group works, first we'll input the point numbers from the illustration below, using **"PIN"**.



keystroke:

1 [ ] STO

prompt: **N1=?**

keystrokes:

1 0 0 R/S

prompt: **E1=?**

keystrokes:

1 0 0 R/S

keystrokes:

2 [ ] STO

prompt: **N2=?**

keystrokes:

2 0 0 R/S

prompt: **E2=?**

keystrokes:

1 1 5 R/S

keystrokes:

prompt: N3=?

keystrokes:

prompt: E3=?

keystrokes:

keystrokes:

prompt: N4=?

keystrokes:

prompt: E4=?

keystrokes:

We now have all of the points in storage. We'll use these points for the next two examples, starting with an example of **radial** inversing.

Using point #1 as the "setup" point, we may calculate radial ties to each of the other points (FLAG 01 is clear) as follows:

keystrokes:

output:

N1=100.0000
E1=100.0000

*

prompt: **TO?**

keystrokes:

output:

AZ=8.3151
HD=101.1187

N2=200.0000
E2=115.0000

*

prompt: **TO?**

keystrokes:

output:

AZ=82.1405
HD=111.0180

N3=115.0000
E3=210.0000

*

prompt: **TO?**

keystrokes:

output:

AZ=153.2606
HD=67.0820

N4=40.0000
E4=130.0000

*

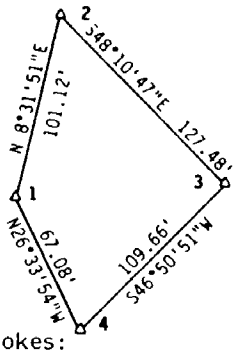
prompt: **TO?**

*An additional  will be needed if used without printer

Keystroke Examples

We'll use the same stored coordinates to do a keystroke example of an **INVERSE TRAVERSE**, using "**XX**". The only difference between this example and the last is that we set FLAG 01 each time, prior to entering the point number.

We will also set FLAG 00 before we start, to have **bearing** output instead of azimuth.



keystrokes:

[ ] 1 [ ] CHS

output:

N1=100.0000
E1=100.0000

* prompt: **TO?**

keystrokes:

[ ] 7 0 0 0
[ ] 7 0 1 2 R/S

output:

N 8.3151 E
HD=101.1187

N2=200.0000
E2=115.0000

* prompt: **TO?**

keystrokes:

[ ] 7 0 1 3 R/S

output:

S 48.1047 E
HD=127.4755

N3=115.0000
E3=210.0000

* prompt: **TO?**

*An additional R/S will be needed if used without printer

keystrokes:

[ ] 7 0 1 4 R/S

output:

S 46.5051 W
HD=109.6586

N4=40.0000
E4=130.0000

* prompt: **TO?**

keystrokes:

[ ] 7 0 1 1 R/S

output:

N 26.3354 W
HD=67.0820

N1=100.0000
E1=100.0000

* prompt: **TO?**

To use "**POUT**", simply input the point number, and stroke [ ] RCL. This recalls the point's coordinates to the x and y registers. The Easting will be displayed, and you may verify that the northing is in the y-register by stroking either X≷Y or R↓.

The time has come to modify our Surveying Pac traverse program. Do a **GTO ..** and shift into **program mode** for a minute, stroking [PRGM]. Check to insure that you have enough available registers (you will need at least 80), then stroke [PRGM] again.

We want to copy the program "**TRAV**" from the Surveying Pac, and this is done by stroking

[XEQ] [ALPHA] [C] [O] [P] [Y] [ALPHA]

at the prompt:

[ALPHA] [T] [R] [A] [V] [ALPHA]

Next, go to program step 001 by stroking

[ ] [RCL] [.] [0] [0] [1]

The display should say **001 LBL TRAV**. Stroke [USER] so that you are not in **user mode** (the shifted "STO" function is assigned to "**PIN**" right now) and type in

[ ] [STO] [ALPHA] [T] [R] [A] [ALPHA] [ ] [SST] [←]

This has renamed the program "**TRA**", and deleted the old name. The revisions to the existing program will be done from the bottom, working upward through the program steps, so that we can keep our program step numbers easy to find. Now, follow these steps in the order shown:

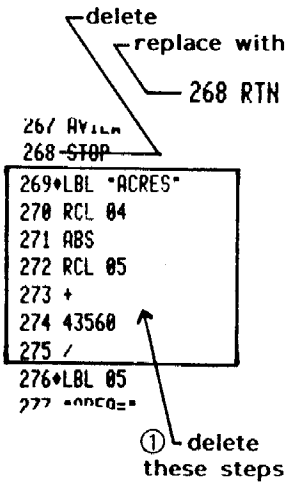
1. Stroke

[ ] [RCL] [.] [2] [7] [5]

to go to program step 275. Use the [←] key to delete steps 275 thru 268, and type in [ ] [EEX] to have RTN at step 268.

2. Go to step 208 by stroking

[ ] [RCL] [.] [2] [0] [8]



Editing the Surveying Pac

```
196 RCL 05
191 +
192 XEQ 05
193 RDN
194 RCL 13
195 ENTER↑
196 CF 01
197 CLA
198 "+ CORR"
199 ASTO 13
200 SF 10
201 XROM "NE"
202 "CLOSURE"
203 R↑
204 STO 13
205 RDN
206 AVIEW
207 XEQ a
208 SF 01
209 STOP
```

② delete these steps

then replace them with these

```
193 "CLSR PT?"
194 PROMPT
195 SF 10
196 STO 18
197 XEQ "PIN"
198 XEQ "OUT"
199 DSE 13
200 ADV
201 CF 01
202 XEQ a
203 SF 01
204 RCL 13
```

③ delete

replace with

168 XROM "INVERSE"

```
.00 RTN
167◆LBL a
168◆LBL "INVERSE"
169 RCL 08
170 -
171 RCL 03
172 -
173 X<>Y
174 RCL 07
175 -
176 RCL 02
177 -
178 XEQ 08
179 RCL 10
180 RCL 01
181 XEQ 04
182 RTN
```

3. Go to step 181 and delete program steps 181 thru 168. Type in **XEQ "INVERSE"**; backstep, and add these steps

```
168 1
169 ST- 13
170 RDN
```

4. Go to program step number 157 (RTN) and delete all of the program steps thru step 102 (RCL 00), backstep and delete step number 100 (LBL "TS").

Singlestep once (the display should now show **100 LBL D**) and type in these steps:

```
101 XROM "TS"
102 XEQ "IN"
103 RTN
```

④ **delete these steps**

```
99 X<>Y
100♦LBL "TS"
101♦LBL D
102 RCL 00
103 X<>Y
104 FS? 01
105 GTO 04
106 RCL 10
107 X<>Y
  ?♦LBL 04
     ?? 01

151 [illegible]
152 FS?C 04
153 SF 01
154 XROM "NE"
155 AVIEW
156 ADV
157 RTN
158♦LBL 10
159 "L="
```

5. Go to step number 042 (CF 01). We want to insert the steps shown to the left between steps 42 and 43 ("SS").

⑤ **insert these steps**

```
43 "PT. NO.?"
44 PROMPT
45 1
46 -
47 STO 13
```

here

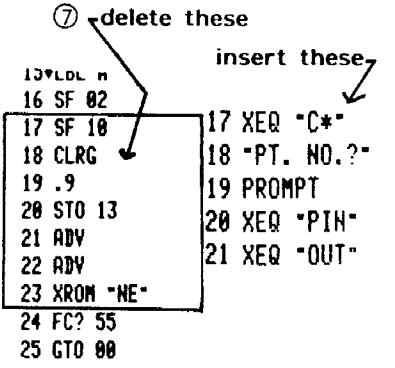
```
40 RTN
41♦LBL J
42 CF 01
43 "SS"
44 AVIEW
```

6. If you Single Step ahead as a check, the next step should now read **48 "SS"**. Step 49 should be **AVIEW**, and step 50 should be **RTN**.

Editing the Surveying Pac

7. Backstep to step 023 (XROM "NE"), delete all of the program steps thru step 17 (SF 10).

Type in the new steps shown to the right.



8. Go to step 010 (LBL 12) and delete it. ~~10*LBL 12~~

Go to the end of the program. As a check, this should be program step 207 END. If it isn't, recheck the steps so far, and correct any errors.

Backstep to program step 206 and type in the following program steps:

```
207 RTN
208*LBL "INV"
209 RCL 08
210 -
211 X<>Y
212 RCL 07
213 -
214 R-P
215 "HD="
216 ARCL X
217 AVIEW
218 X<>Y
219 HMS
220 XROM "BRG"
221 AVIEW
222 ADV
223 RTN
224*LBL "CO"
225 0
226 STO 11
227 STO 14
228 STO 15
229 "BEG. PT."
230 PROMPT
231 XEQ "POUT"
232 XROM "NE"
233 AVIEW
234 ADV
235 1
236 ST+ 13
```

```
237 RCL 10
238 RCL 01
239 CHS
240 P-R
241 RCL 06
242 /
243 CHS
244 STO 16
245 X<>Y
246 RCL 06
247 /
248 CHS
249 STO 05
250♦LBL 02
251 RCL 13
252 RCL 18
253 X=Y?
254 GTO 01
255 XEQ "OUT"
256 STO 01
257 STO 15
258 X<>Y
259 STO 00
260 STO 14
261 X<>Y
262 RCL 08
263 -
264 X<>Y
265 RCL 07
266 -
267 R-P
268 ST+ 11
269 RCL 11
270 RCL 16
271 *
272 ST+ 14
273 RCL 11
274 RCL 05
275 *
276 ST+ 15
277 RCL 14
278 RCL 15
279 XEQ "IN"
280 RCL 13
281 1
282 -
283 XEQ "POUT"
284 STO 08
285 X<>Y
286 STO 07
287 1
288 ST+ 13
289 XEQ "OUT"
290 XEQ "INV"
291 RCL 00
292 STO 07
293 RCL 01
294 STO 08
295 XEQ "OUT"
296 XROM "NE"
297 AVIEW
298 ISG 13
299 ADV
300 ADV
301 GTO 02
302 RTN
303♦LBL 01
304 STOP
305 RTN
306 END
```

Go to "XX". Go to step 065 (END), and delete it. Now all of the programs are combined, and may be used with each other from the keyboard.

Traversing

Before you try out your new program, you need to re-size the calculator to establish enough storage registers for your coordinates.

With an otherwise empty calculator memory (only "**TRA**" and the utilities in program memory) you can store 100 coordinate pairs. The storage registers begin at register 20 and two are needed for each coordinate pair.

One easy way to check storage capacity is to size the calculator to 020, pack with a **GTO ••.** and stroke PRGM The number of registers left, divided by two, is the number of points you may store.

Resize the calculator to the number of registers shown after packing, divided by two, +18.

The program steps are exactly the same as shown on pages 10 through 28 of the HP Surveying Pac instruction manual, with two exceptions. The first of these is an added prompt following the **DSP BRG?** and **DSP L/D?** prompts. The added prompt is "**PT. NO.?**

This prompt is added for insurance against overwriting any coordinates you may already have in storage. The second difference is in the closure routine, LBL E, and we'll do a short example traverse before we look at that. Otherwise, you won't notice any real difference (it runs a little faster), but it's storing the coordinates as it calculates them.

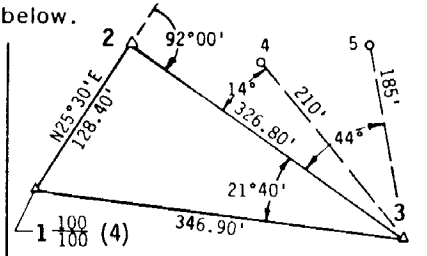
Begin with XEQ ALPHA T R A ALPHA and we'll work the little traverse shown below.

prompt: **DSP BRG?**

keystrokes: Y R/S

prompt: **DSP L/D?**

keystrokes: N R/S



prompt: PT. NO.?

keystrokes:

1 R/S

prompt: N1=?

keystrokes:

1 0 0 R/S

prompt: E1=?

keystrokes:

1 0 0 R/S

output:

N1=100.0000
E1=100.0000 *

keystrokes:

2 5 . 3 B

prompt: QD=?

keystrokes:

1 R/S

output:

N 25.3000 E

keystrokes:

1 2 8 . 4 D

output:

HD=128.4000

N2=215.8920
E2=155.2776 *

keystrokes:

9 2 [illegible] C

output:

S 62.3000 E

keystrokes:

3 2 6 . 8 D

output:

HD=326.8000

N3=64.9925
E3=445.1528 *

keystrokes:

2 1 . 4 CHS C

output:

N 84.1000 W

keystrokes:

3 4 6 . 9 D

output:

HD=346.9000

N4=100.2497
E4=100.0491 *

Now for the closure: Again the added prompt for point number. Input the next highest number.

keystrokes:

E

output:

ΣHD=802.1000
AREA=20,923.8192

prompt: CLSR PT?

keystrokes:

5 R/S

prompt: N5=?

keystrokes:

1 0 0 R/S

prompt: E5=?

keystrokes:

1 0 0 R/S

output:

S 11.0727 W
HD=0.2545

N4=100.0000
E4=100.0000

*An additional R/S will be needed if used without printer

Compass Correction

There are actually two differences in the closure routine, compared to the Surveying Pac. The first was the added prompt for a closure point number, and the second is that the closure isn't labeled. The closing to one point number higher than the actual point is the same in the Surveying Pac. There, the number is generated by the program.

If you leave these coordinates in storage and begin another traverse, or want to do some sideshots, you can begin with point number 5, since it is a duplicate point of number one. You could also re-use point number 4, after the traverse has been adjusted, since it will then be equal to point number 1.

A couple of words of caution: Most of the routines in the Surveying Pac begin with a sequence which includes **CLRG**, the command to clear the registers. If you have to use a HP routine, it is a good idea to protect your points by storing them.

At this point, if you want to use the built-in compass correction routine, the procedure is:

stroke [XEQ] [ALPHA] [C] [O] [ALPHA]

prompt: **BEG PT?**

keystrokes:

[1] [R/S]

output: the output is shown to the right, and is automatic. The program will calculate the adjusted coordinates from storage, inversing each course before output of the coordinates.

```
N1=100.0000
E1=100.0000

HD=128.3605
N 25.3016 E

N2=215.8520
E2=155.2698

HD=326.8292
S 62.2857 E

N3=64.8508
E3=445.1249

HD=346.9102
N 84.1105 W

N4=100.0000
E4=100.0000
```

The procedure which we recommend for traversing with this program is

1. Run the raw traverse through the program, ignoring sideshots.

2. Adjust the coordinates (the adjusted coordinates are stored in place of the original ones).

3. Rerun the traverse, using the adjusted information, and do the sideshots as you go.

That last isn't as difficult as it sounds, because we can inverse traverse by point number. To do the sideshots, execute "**TRA**" and answer the first few prompts, through the input of the starting coordinates. Recall the second point (#2 in the example) by executing "**POUT**", and inverse to it.

.... **output:** N1=100.0000
E1=100.0000

keystrokes:

2 [ ] RCL

pause

[ ] A

output: N 25.3816 E
HD=128.3605

N2=215.8520
E2=155.2698

Repeat the process, and inverse to point 3
keystrokes:

3 [ ] RCL

pause

[ ] A

output: S 62.2857 E
HD=326.8292

N3=64.8508
E3=445.1249

keystroke: J

prompt: **PT. NO.?**

keystrokes:

4 R/S

output: SS

keystrokes:

1 4 C

output: N 48.2857 W

keystrokes:

2 1 0 D

output: HD=210.0000

N4=204.0489
E4=287.8866 *

keystroke: J

prompt: **PT. NO.?**

keystrokes:

5 R/S

output: SS

keystrokes:

4 4 C

output: N 18.2857 W

keystrokes:

1 8 5 D

output: HD=185.0000

N5=240.3085
E5=386.4769

keystroke: I

output: TRAV

Resume inversing to the next point (in this case point number 1).

*An additional R/S will be needed if used without printer

Appendix A

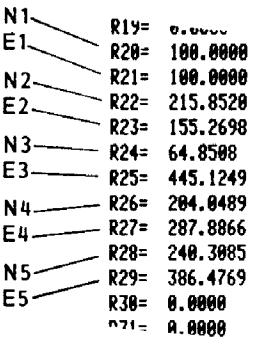
The storage registers of any point may be found as follows:

north coordinate

18 + 2(PT#)

east coordinate

19 + 2(PT#)



To store data onto a magnetic card, place **20.eee** (where eee is the three-digit number of the highest register) and execute WDTAX.

If your edition of this book includes the program cards, the program is carried on tracks 1 and 2 of cards A, B and C, and on track 1 of card D.

To help with the "proof reading" chores after editing, a complete program listing of the final form of the program is included below.

```
01♦LBL "XX"
02 XROM "*IN"
03 XEQ "C*"
04 CF 10
05 XEQ "POUT"
06 XROM "NE"
07 AVIEW
08 ADV
09 STO 08
10 RDN
11 STO 07
12♦LBL 03
13 CF 01
14 "TO?"
15 PROMPT
16 XEQ "POUT"
17 DSE 13
18 STO X
19 XROM "INVERSE"
20 GTO 03
21 RTN
22♦LBL "POUT"
23 STO 13
24♦LBL "OUT"
25 RCL 13
26 2
27 *
28 19
29 +
30 ENTER↑
31 ENTER↑
32 1
33 -
34 RCL IND X
35 RCL IND Z
36 RTN
37♦LBL "PIN"
38 SF 10
39 STO 13
40 XROM "NE"
41♦LBL "IN"
42 RCL 13
43 2
44 *
45 19
46 +
47 X<>Y
48 STO IND Y
    `DN
    1
51 -
52 X<>Y
53 STO IND Y
54 RTN
55♦LBL "C*"
56 .019
57 0
58♦LBL 11
59 STO IND Y
60 ISG Y
61 GTO 11
62 RDN
63 RDN
64 RTN
65♦LBL "TRA"
66 XROM "*IN"
67 CF 05
68 SF 01
69 SF 00
70 "DSP BRG"
71 XROM "*YN"
72 FC? 10
73 CF 00
74 "DSP L/D"
75 XROM "*YN"
76 FS? 10
77 SF 03
78♦LBL A
79 SF 02
80 XEQ "C*"
81 "PT. NO.?"
82 PROMPT
83 XEQ "PIN"
84 XEQ "OUT"
85 FC? 55
86 GTO 00
87 CF 10
88 XROM "NE"
89 AVIEW
90 ADV
91♦LBL 00
92 STO 08
93 RDN
94 STO 07
95 STOP
96♦LBL I
97 SF 01
98 CF 02
99 "TRAV"
100 AVIEW
101 RTN
102♦LBL J
103 CF 01
104 "PT. NO.?"
105 PROMPT
106 1
107 -
108 STO 13
109 "SS"
110 AVIEW
111 RTN
112♦LBL H
113 SF 05
114♦LBL C
115 PI
116 R-D
117 HMS+
118 FS?C 05
119 GTO b
120♦LBL c
```

```
121 RCL 00
122 HMS
123 HMS+
124 GTO b
125*LBL B
126 SF 10
127 XROM "*A1"
128 HMS
129*LBL b
130 HR
131 1
132 P-R
133*LBL 08
134 CF 10
135 R-P
136 STO 01
137 X<>Y
138 X>0?
139 GTO 07
140 360
141 +
142*LBL 07
143 FS?C 02
144 STO 00
145 FS? 01
146 STO 00
147 STO 10
148 FC? 00
149 GTO 09
150 XROM "*B1"
151 AVIEW
152 RTN
153*LBL 09
154 HMS
155 XROM "*A0"
156 AVIEW
157 RTN
158*LBL d
159 "x=?"
160 PROMPT
161 HR
162 X<>Y
163 P-R
164 X<=Y?
165 X<>Y
166*LBL D
167 XROM "TS"
168 XEQ "IN"
169 RTN
170*LBL 10
171 "L="
172 ARCL X
173 AVIEW
174 "D="
175 ARCL Y
176 AVIEW
177 ADV
178 RTN
179*LBL a
180 1
181 ST- 13
182 RDN
183 XROM "INVERSE"
184 RTN
185*LBL E
186 SF 04
187 "ΣHD="
188 ARCL 06
189 AVIEW
190 RCL 04
191 ABS
192 RCL 05
193 +
194 XEQ 05
195 "CLSR PT?"
196 PROMPT
197 SF 10
198 STO 10
199 XEQ "PIN"
200 XEQ "OUT"
201 DSE 13
202 ADV
203 CF 01
204 XEQ a
205 SF 01
206 RCL 13
207 STOP
208*LBL e
209 SF 10
210 XROM "*BL"
211 CF 10
212 HR
213 STO 12
214 SIN
215 LASTX
216 D-R
217 STO 14
218 -
219 "R=?"
220 PROMPT
221 STO 11
222 SIGN
223 X<>Y
224 CHS
225 LASTX
226 X↑2
227 *
228 2
229 /
230 *
231 ST+ 05
232 "SEG="
233 ARCL X
234 AVIEW
235 RCL 14
236 RCL 11
237 *
238 ABS
239 ST+ 06
240 "L="
241 ARCL X
242 AVIEW
243 RCL 12
244 2
245 /
246 TAN
247 RCL 11
248 *
249 ABS
250 "T="
251 ARCL X
252 AVIEW
253 RCL 12
254 2
255 /
256 SIN
257 RCL 11
258 *
259 2
260 *
261 ABS
262 ST- 06
263 "C="
264 ARCL X
265 AVIEW
266 RTN
267*LBL 05
268 "AREA="
269 ARCL X
270 AVIEW
271 RTN
272*LBL "INV"
273 RCL 08
274 -
275 X<>Y
276 RCL 07
277 -
278 R-P
279 "HD="
280 ARCL X
281 AVIEW
282 X<>Y
283 HMS
284 XROM "BRG"
285 AVIEW
286 ADV
287 RTN
288*LBL "CO"
289 0
290 STO 11
291 STO 14
292 STO 15
293 "BEG. PT."
294 PROMPT
295 XEQ "POUT"
296 XROM "NE"
297 AVIEW
298 ADV
299 1
300 ST+ 13
301 RCL 10
302 RCL 01
303 CHS
304 P-R
305 RCL 06
306 /
307 CHS
308 STO 16
309 X<>Y
310 RCL 06
311 /
312 CHS
313 STO 05
314*LBL 02
315 RCL 13
316 RCL 18
317 X=Y?
318 GTO 01
319 XEQ "OUT"
320 STO 01
321 STO 15
322 X<>Y
323 STO 00
324 STO 14
325 X<>Y
326 RCL 08
327 -
328 X<>Y
329 RCL 07
330 -
331 R-P
332 ST+ 11
333 RCL 11
334 RCL 16
335 *
336 ST+ 14
337 RCL 11
338 RCL 05
339 *
340 ST+ 15
341 RCL 14
342 RCL 15
343 XEQ "IN"
344 RCL 13
345 1
346 -
347 XEQ "POUT"
348 STO 08
349 X<>Y
350 STO 07
351 1
352 ST+ 13
353 XEQ "OUT"
354 XEQ "INV"
355 RCL 00
356 STO 07
357 RCL 01
358 STO 08
359 XEQ "OUT"
360 XROM "NE"
361 AVIEW
362 ISG 13
363 ADV
364 ADV
365 GTO 02
366 RTN
367*LBL 01
368 STOP
369 RTN
370 END
```