

การทดสอบโปรแกรม



4.1 การทดสอบโปรแกรมและตัวอย่าง

4.1.1 การทดสอบโปรแกรม

ทำตามขั้นตอนต่อไปนี้

1) กดปุ่ม SELFTEST ของเครื่องพลอตเตอร์ พลอตเตอร์จะทำงานตามคำสั่งชุดหนึ่งซึ่งประกอบด้วยคำสั่งทุกคำสั่งที่มีอยู่ในพลอตเตอร์ซีพี-01 คำสั่งชุดนี้ประกอบด้วยคำสั่งดังต่อไปนี้

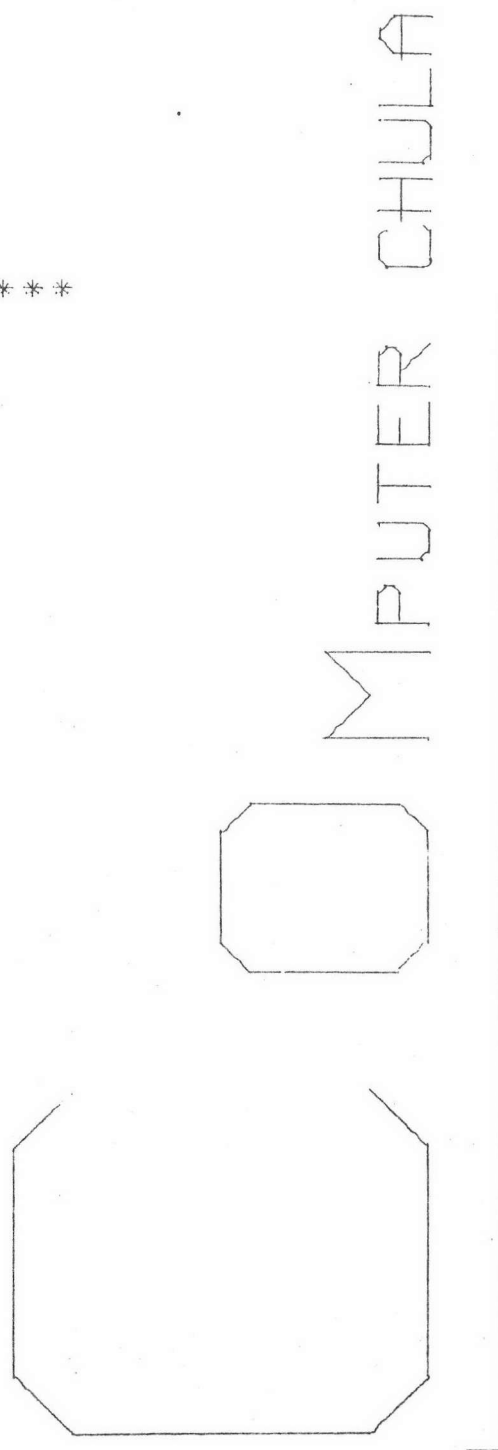
```
"PU MO PRO200,0200 S5 CH_C_S4 CH_O_S3 CH_M_S2  
CH_PUTER CHULA_ PA0200,0150 PD MR0800 PU ML0800 PD MU0025 PU MDO025  
PU PA1000,0500 S1 CV_***_MO"
```

รูปที่ 4.1 แสดงการทำงานของพลอตเตอร์เมื่อกดปุ่ม SELFTEST ผลปรากฏว่าพลอตเตอร์ทำงานตามคำสั่งชุดนี้ถูกต้องแสดงว่าโปรแกรมควบคุมสามารถแปลคำสั่งได้ถูกต้องและอยู่ในสภาพพร้อมที่จะทำงาน

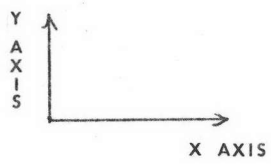
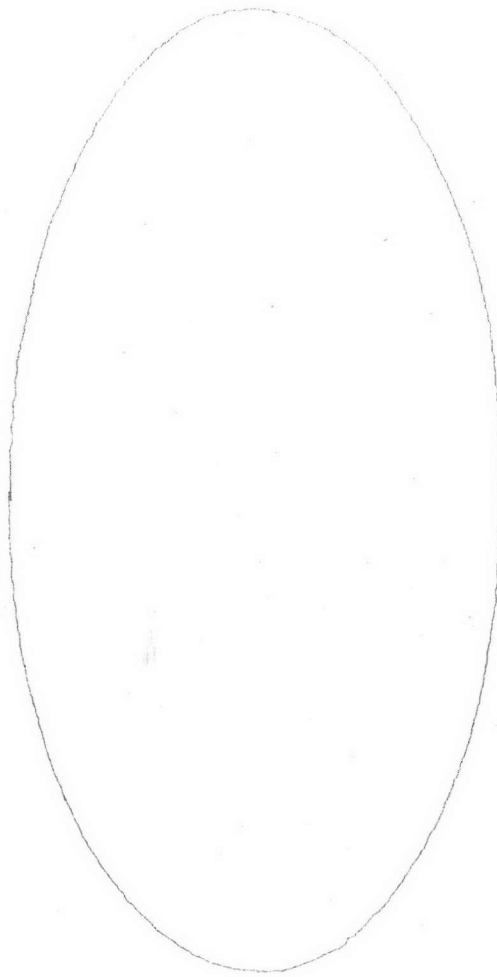
2) ทดลองส่งคำสั่งต่าง ๆ เป็นรหัส ASCII จากไมโครคอมพิวเตอร์ไปยังพลอตเตอร์ ผลปรากฏว่า พลอตเตอร์ทำงานถูกต้องแสดงว่าการส่งข้อมูลจากไมโครคอมพิวเตอร์ไปยังพลอตเตอร์เป็นไปอย่างถูกต้อง

3) ทดลองส่งคำสั่ง 1 คำสั่งจากไมโครคอมพิวเตอร์ไปยังพลอตเตอร์ แล้วคอยรับข่าวสารจากพลอตเตอร์หลังจากพลอตเตอร์ทำงานตามคำสั่งนั้นเสร็จสิ้นแล้ว ผลปรากฏว่าได้รับข่าวสารนั้นแสดงว่าการรับข้อมูลที่ส่งมาจากพลอตเตอร์มายังไมโครคอมพิวเตอร์เป็นไปอย่างถูกต้อง

4) ทดลองส่งคำสั่ง PA จำนวน 128 คำสั่งซึ่งมีขนาด 1.5 กิโลไบต์ไปยังพลอตเตอร์ ผลปรากฏว่า พลอตเตอร์ทำงานตามคำสั่งได้ถูกต้องทั้งหมด แสดงว่าการขยายขนาดของบัพเฟอร์เป็น 1.5 กิโลไบต์เป็นไปอย่างถูกต้อง



รูปที่ 4.1 แสดงการทำงานของพลอตเตอร์เมื่อกดปุ่ม SELFTEST



รูปที่ 4.2 แสดงการทำงานของฟลोटเตอร์

5) ทดลองเขียนโปรแกรมภาษาเบสิกโดยเรียกใช้โปรแกรมย่อยต่าง ๆ ผลปรากฏว่า พลอตเตอร์ทำงานได้ถูกต้องตามที่ต้องการดังรูปที่ 4.2 แสดงว่าโปรแกรมย่อย ภาษาเบสิกทำงานถูกต้อง

4.1.2 ตัวอย่างโปรแกรม

4.1.2.1 ตัวอย่างโปรแกรมการสร้างกราฟรูป SINE

โปรแกรมนี้เป็นการสร้างกราฟของสมการ

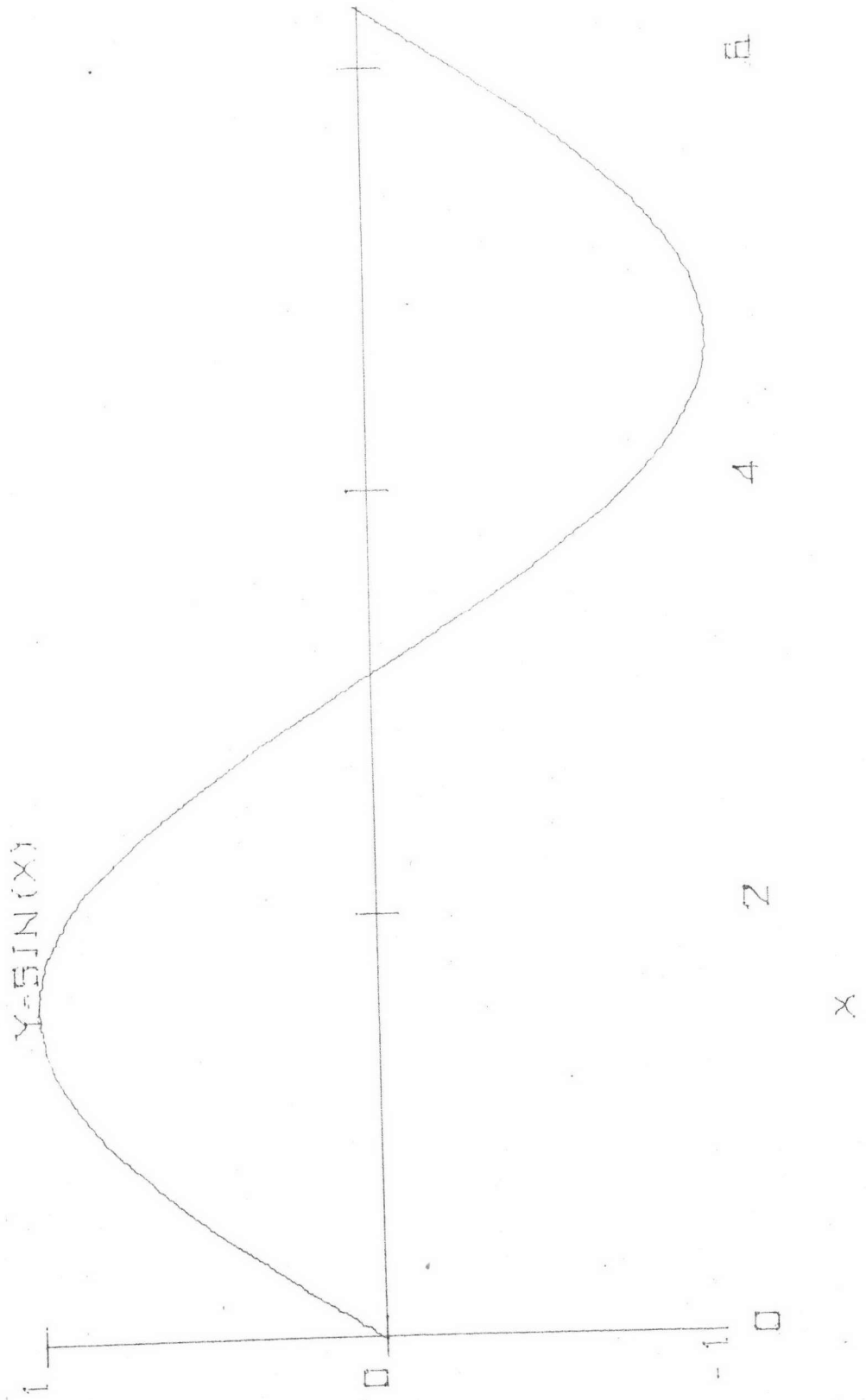
$$y = \sin(x)$$

$$0 \leq x \leq 2\pi$$

```

10 *SINEWAVE
20 PRINT CHR$(26)
30 GOSUB 10000
40 X. LENGTH=8
50 Y. LENGTH=4
60 X. MIN=0
70 X. MAX=6.3
80 Y. MIN=-1
90 Y. MAX=1
100 GOSUB 11000
110 X. TICK=2
120 Y. TICK=1
130 X. TITLE$="X": Y. TITLE$="Y": M. TITLE$="Y=SIN(X)": GOSUB 12000
140 GOSUB 19000
150 X. VALUE=0: Y. VALUE=SIN(X)
160 GOSUB 14000
170 PRINT
180 FOR X = 0 TO 6.3 STEP .1
190 Y=SIN(X)
200 X. VALUE=X
210 Y. VALUE=Y
220 GOSUB 15000
230 NEXT X
240 GOSUB 19000
250 PRINT : PRINT "ALLRIGHT"
260 END

```



รูปที่ 4.3 แสดงกราฟรูป SINE

4.1.2.1 ตัวอย่างโปรแกรมการสร้างกราฟฟังก์ชันเอกซ์โพเนนเชียล

(Exponential)

โปรแกรมนี้เป็นโปรแกรมการสร้างกราฟของฟังก์ชัน

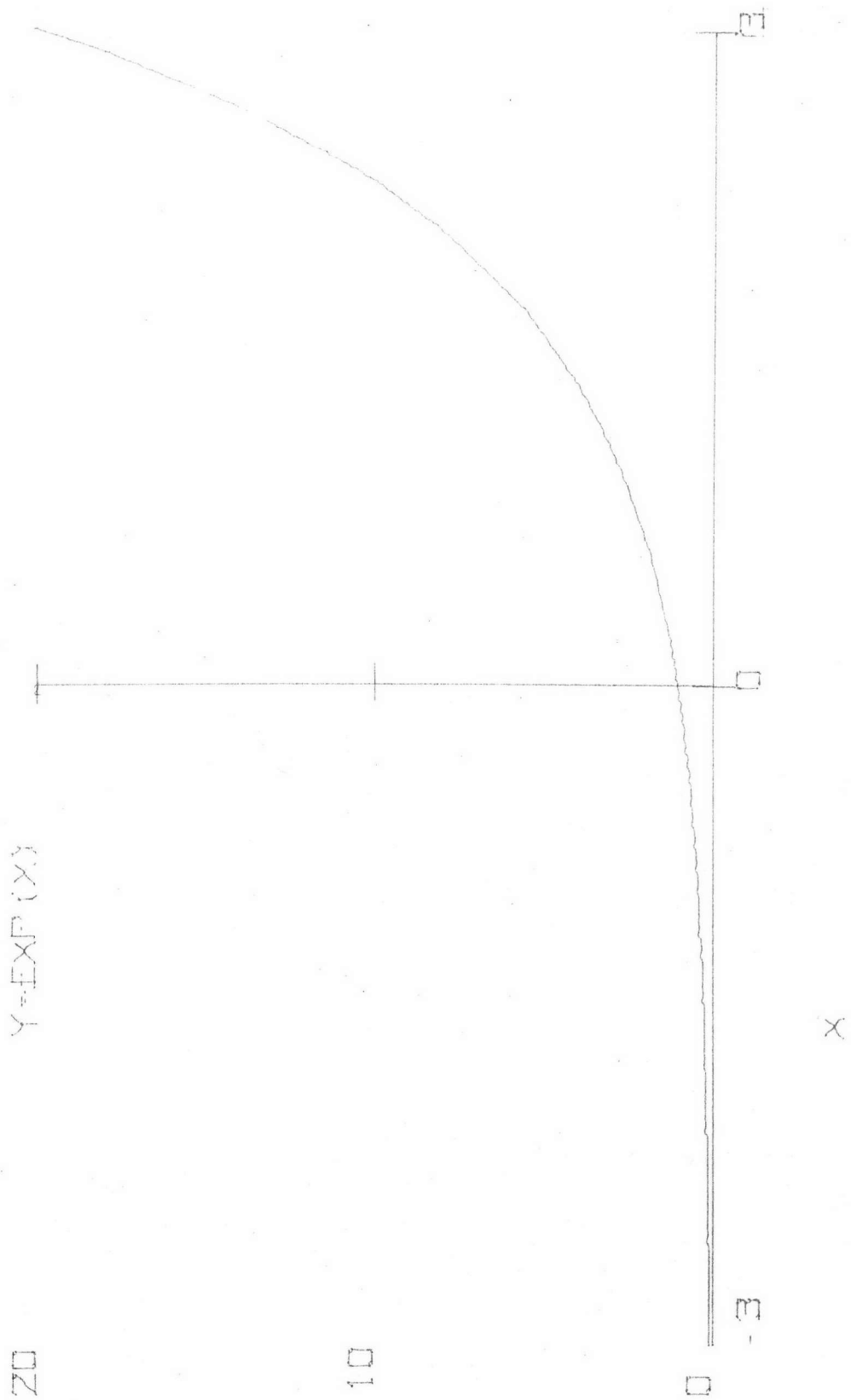
$$y = e^x$$

$$-3 \leq x \leq 3$$

```

10 'EXPONENTIAL
20 PRINT CHR$(26)
30 GOSUB 10000
40 X.LENGTH=8
50 Y.LENGTH=4
60 X.MIN=-3
70 X.MAX=3
80 Y.MIN=0
90 Y.MAX=20
100 GOSUB 11000
110 X.TICK=3
120 Y.TICK=10
130 X.TITLE$="X":Y.TITLE$="Y":M.TITLE$="Y=EXP(X)":GOSUB 12000
140 GOSUB 19000
150 X.VALUE=-3:Y.VALUE=EXP(X.VALUE)
160 GOSUB 14000
170 PRINT
180 FOR X = -3 TO 3 STEP .1
190 Y=EXP(X)
200 X.VALUE=X
210 Y.VALUE=Y
220 GOSUB 15000
230 NEXT X
240 GOSUB 19000
250 PRINT : PRINT "ALLRIGHT"
260 END

```



รูปที่ 4.4 แสดงกราฟฟังก์ชันเอกซ์โพเนนเชียล

4.1.2.3 ตัวอย่างโปรแกรมการสร้างกราฟเส้นตรง กราฟแท่ง และ

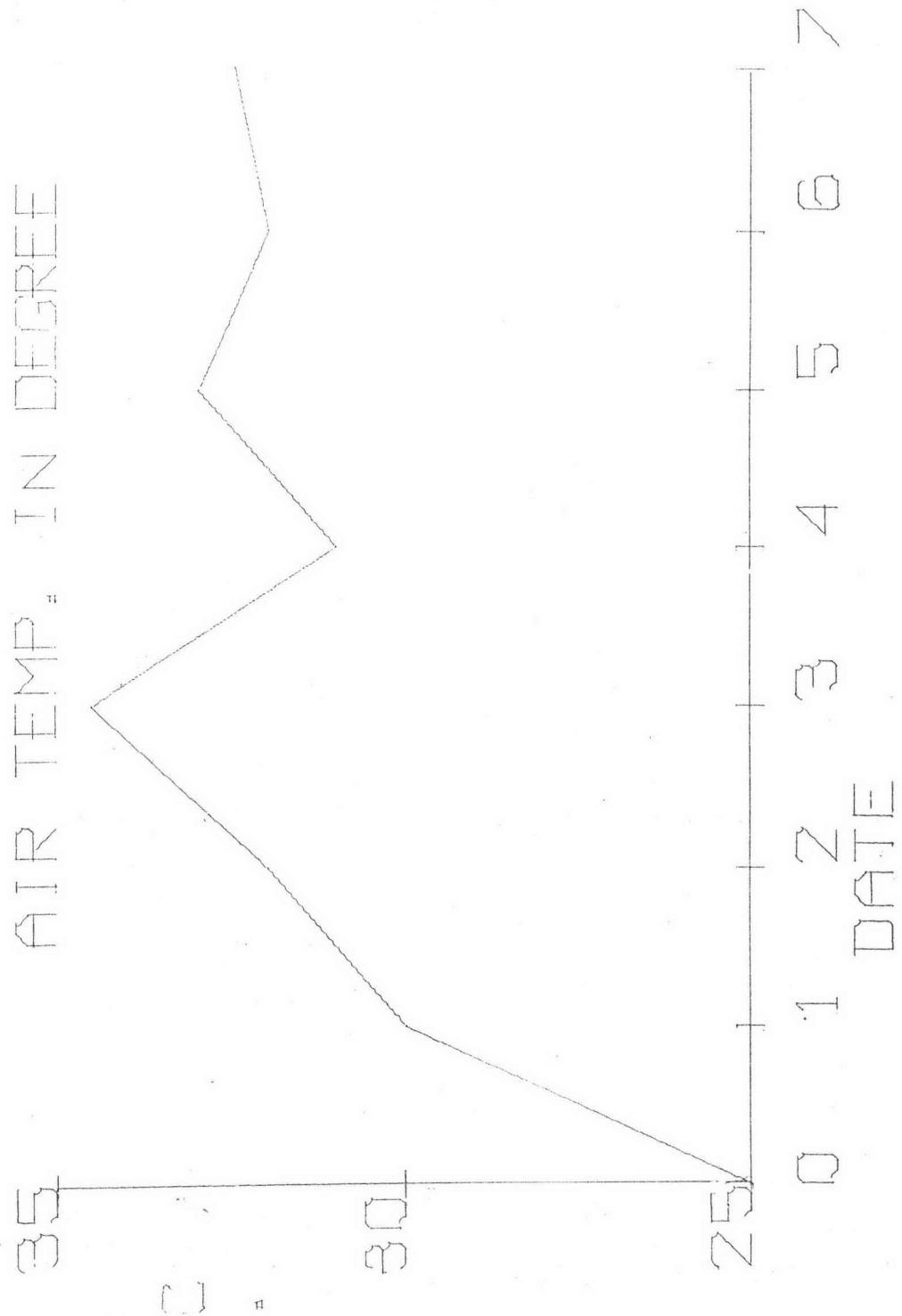
กราฟวงกลม


```
10 /
20 /AUTO PLOT PROGRAM FOR INTERACTIVE PLOTTING
30 TAB1=30
40 PRINT CHR$(26)
50 PRINT:PRINT:PRINT
60 PRINT TAB(24); "CP-01 PLOTTER AUTO PLOT PROGRAM"
70 PRINT:PRINT
80 PRINT TAB(30); "DO YOU WANT ... "
90 PRINT:PRINT
100 PRINT TAB(TAB1); "1. LINE GRAPH"
110 PRINT TAB(TAB1); "2. BAR CHART"
120 PRINT TAB(TAB1); "3. PIE CHART"
130 PRINT:PRINT:PRINT
140 PRINT TAB(30); "PLEASE SELECT (1-3)";
150 INPUT FLAG
160 ON FLAG GOSUB 500,1000,2000
170 PRINT CHR$(26); "GOOD BYE....."
180 END
500 /
510 / LINE
520 PRINT CHR$(24),CHR$(26)
530 PRINT TAB(30); "LINE GRAPH":PRINT
540 INPUT "X AXIS LENGTH...? ", X.LENGTH
550 INPUT "Y AXIS LENGTH...? ", Y.LENGTH
560 INPUT "MINIMUM X VALUE...? ", X.MIN
570 INPUT "MAXIMUM X VALUE...? ", X.MAX
580 INPUT "MINIMUM Y VALUE...? ", Y.MIN
590 INPUT "MAXIMUM Y VALUE...? ", Y.MAX
600 INPUT "X TICK ...? ", X.TICK
610 INPUT "Y TICK ...? ", Y.TICK
620 INPUT "DATA COUNT...? ", COUNT
630 INPUT "X TITLE: ", X.TITLE$
640 INPUT "Y TITLE: ", Y.TITLE$
650 INPUT "MAIN TITLE: ", M.TITLE$
660 GOSUB 10000
670 GOSUB 11000
680 GOSUB 12000
690 GOSUB 19000
700 X.VALUE=X.MIN:Y.VALUE=Y.MIN
710 GOSUB 14000
720 GOSUB 10170
730 PRINT
740 FOR DDD= 1 TO COUNT
750 INPUT "X, Y...?", X, Y
760 X.VALUE=X:Y.VALUE=Y
770 GOSUB 15000
780 GOSUB 10170
790 NEXT DDD
800 GOSUB 19000
810 PRINT : PRINT "ALLRIGHT"
820 RETURN
830 /END OF LINE
840 /
```

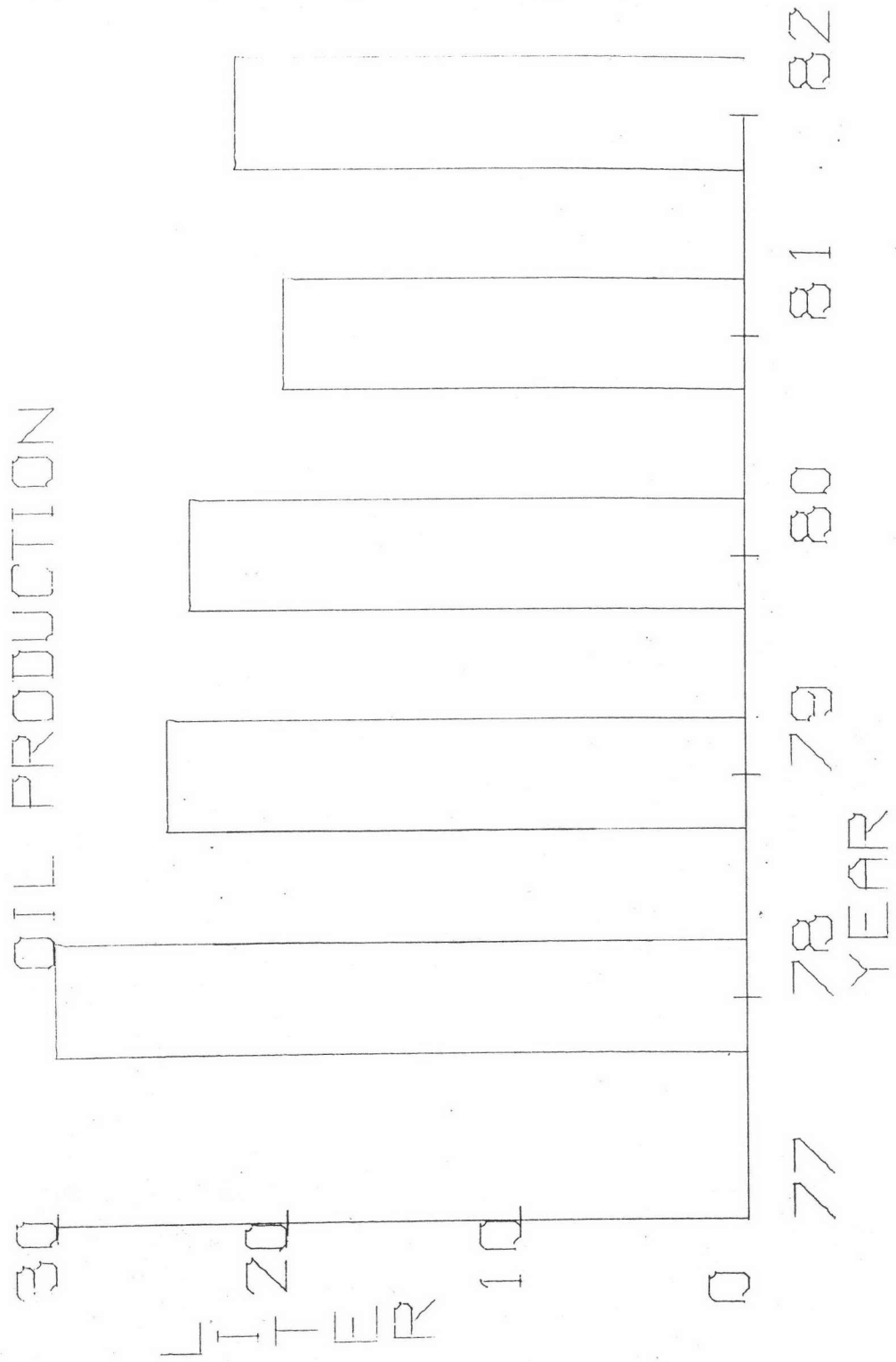
```
1000 /
1010 /BAR CHART
1020 PRINT CHR$(24),CHR$(26)
1030 PRINT TAB(30);"BAR CHART":PRINT
1040 INPUT "X AXIS LENGTH...? ",X.LENGTH
1050 INPUT "Y AXIS LENGTH...? ",Y.LENGTH
1060 INPUT "MINIMUM X VALUE...? ",X.MIN
1070 INPUT "MAXIMUM X VALUE...? ",X.MAX
1080 INPUT "MINIMUM Y VALUE...? ",Y.MIN
1090 INPUT "MAXIMUM Y VALUE...? ",Y.MAX
1100 INPUT"X.TICK...? ",X.TICK
1110 INPUT"Y.TICK...? ",Y.TICK
1120 INPUT "BAR WIDTH...? ",BAR.WIDTH
1130 INPUT"DATA COUNT...? ",COUNT
1140 INPUT"X TITLE: ",X.TITLE$
1150 INPUT"Y TITLE: ",Y.TITLE$
1160 INPUT"MAIN TITLE: ",M.TITLE$
1170 GOSUB 10000
1180 GOSUB 11000
1190 GOSUB 12000
1200 GOSUB 19000
1210 X.VALUE=X.MIN:Y.VALUE=Y.MIN:GOSUB 15000
1220 X.VALUE=(XDIFF.P/COUNT)-(BAR.WIDTH/2)
1230 Y.VALUE=0
1240 GOSUB 16000
1250 GOSUB 20200 /PD
1260 GOSUB 10170
1270 PRINT
1280 FOR DDD= 1 TO COUNT
1290 INPUT "Y..?",Y:X.VALUE =0:Y.VALUE=Y-Y.MIN
1300 GOSUB 16000
1310 X.VALUE=BAR.WIDTH:Y.VALUE=0
1320 GOSUB 16000
1330 X.VALUE=0:Y.VALUE=-(Y-Y.MIN)
1340 GOSUB 16000
1350 X.VALUE=((XDIFF.P/COUNT)-BAR.WIDTH):Y.VALUE=0
1360 GOSUB 20000
1370 GOSUB 16000
1380 GOSUB 20200
1390 GOSUB 10170
1400 NEXT DDD
1410 GOSUB 19000
1420 PRINT : PRINT "ALLRIGHT"
1430 RETURN
1440 /END OF BAR CHART
1450 /
```

```
1460 / PIE CHART
2000 PRINT CHR$(26)
2010 PRINT TAB(30); "PIE CHART": PRINT
2020 INPUT "X AXIS LENGTH ", X
2030 INPUT "Y AXIS LENGTH ", Y
2040 X. TICK=1: Y. TICK=1
2050 RADIUS=(Y-2)/2
2060 X. VALUE=RADIUS+1
2070 Y. VALUE=RADIUS+1: XCENTER=X. VALUE: YCENTER=Y. VALUE
2080 INPUT "HOW MANY SECTORS ? ", N
2090 PRINT: PRINT: PRINT "PLEASE WAIT. . . . "
2100 X. LENGTH=14: Y. LENGTH=8: X. MIN=0: X. MAX=14:
Y. MIN=0: Y. MAX=8
2110 GOSUB 11000
2120 GOSUB 18000
2130 GOSUB 19000
2140 DIM ITEM(N)
2150 DIM N$(N)
2160 INPUT "MAIN TITLE: ", CSTRING$
2170 X. VALUE=1: Y. VALUE=. 5: SIZE=2: DIRECTION=1
2180 GOSUB 17000
2190 GOSUB 19000
2200 IF RADIUS<2 THEN SIZE=1
2210 FOR I = 1 TO N
2220 PRINT "SECTOR NO. "; I;
2230 INPUT "NAME: ", N$(I)
2240 INPUT "VALUE = ", ITEM(I)
2250 PRINT
2260 TOTAL=TOTAL+ITEM(I)
2270 NEXT
2280 THETA=0
2290 X. VALUE=XCENTER-(MOVE. P/100): Y. VALUE=YCENTER-(MOVE. P/100)
2300 GOSUB 14000
2310 X. VALUE=RADIUS+XCENTER-(MOVE. P/100): Y. VALUE=YCENTER-(MOVE. P/100)
2320 GOSUB 15000
2330 FOR I = 1 TO N-1
2340 X. VALUE=XCENTER-(MOVE. P/100)
2350 Y. VALUE=YCENTER-(MOVE. P/100)
2360 GOSUB 14000
2370 ANGLE=ITEM(I)/TOTAL*6. 2832
2380 THETA=ANGLE+THETA
2390 X. VALUE=RADIUS*COS(THETA)+XCENTER-(MOVE. P/100)
2400 Y. VALUE=RADIUS*SIN(THETA)+YCENTER-(MOVE. P/100)
2410 GOSUB 15000: GOSUB 20000
2420 THETA2=THETA-(ANGLE/2)
2430 X. VALUE=(RADIUS*COS(THETA2)*. 75)+XCENTER
2440 Y. VALUE=(RADIUS*SIN(THETA2)*. 75)+YCENTER
2450 CSTRING$=RIGHT$(STR$(I), LEN(STR$(I))-1)
2460 DIRECTION=1
2470 GOSUB 17000
2480 NEXT
```

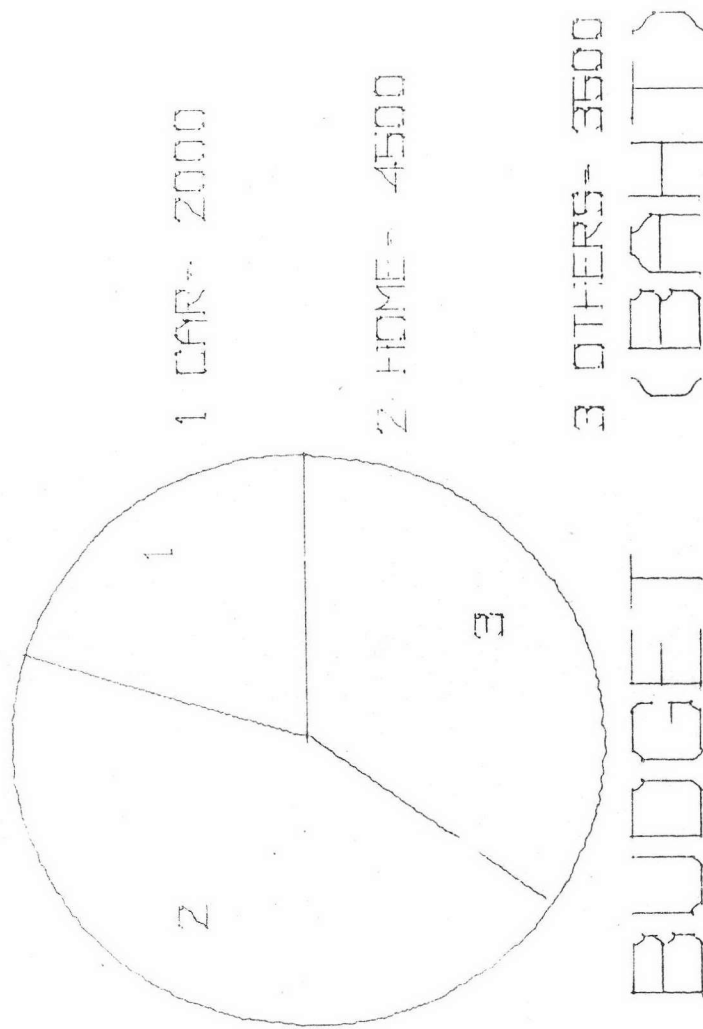
```
2490 THETA2=THETA+((ITEM(N)/TOTAL)*6.2832)/2
2500 X.VALUE=(RADIUS*COS(THETA2)*.75)+XCENTER
2510 Y.VALUE=(RADIUS*SIN(THETA2)*.75)+YCENTER
2520 CSTRING$=RIGHT$(STR$(N),LEN(STR$(N))-1)
2530 GOSUB 17000
2540 GOSUB 19000
2550 FOR I = 1 TO N
2560 X.VALUE=2*RADIUS+I
2570 Y.VALUE=Y-I-I
2580 CSTRING$=STR$(I)+" "+N$(I)+"="+STR$(ITEM(I))
2590 PRINT CSTRING$
2600 DIRECTION=1
2610 GOSUB 17000
2620 PRINT "I", X.VALUE, Y.VALUE
2630 NEXT I
2640 GOSUB 19000
2650 ERASE N$
2660 ERASE ITEM
2670 RETURN
```



รูปที่ 4.5 แสดงกราฟเส้นตรง



รูปที่ 4.6 แสดงกราฟแท่ง



รูปที่ 4.7 แสดงกราฟวงกลม

4.2 วิธีใช้โปรแกรม

การใช้โปรแกรมทำตามขั้นตอนต่อไปนี้

1) ออกแบบโปรแกรมหลักโดยเลือกใช้โปรแกรมย่อยต่าง ๆ ที่มีตามความจำเป็นและความเหมาะสม โปรแกรมหลักต้องมีเลขกำกับคำสั่งน้อยกว่า 10000 และไม่ใช้ชื่อตัวแปรที่ลงท้ายด้วย .P เช่น XXXX .P

2) เปิดเครื่องไมโครคอมพิวเตอร์ให้อยู่ภายใต้ MBASIC Interpreter บ่อนโปรแกรมหลักเข้าสู่หน่วยความจำ

3) ใช้คำสั่ง MERGE "filename" เพื่อนำโปรแกรมย่อยที่ต้องการจากแผ่นบันทึกในดิสก์ไดรฟ์ (Disk Drive) A เข้าสู่หน่วยความจำ โปรแกรมหลักและโปรแกรมย่อยจะถูกรวมเป็นโปรแกรมเดียวกัน

OK

MERGE "INIT"

OK

ถ้าโปรแกรมย่อยอยู่ในแผ่นบันทึกในไดรฟ์ B จะใช้คำสั่ง MERGE "B:filename"

เช่น

OK

MERGE "B: INIT"

OK

4) เมื่อรวมโปรแกรมหลักและโปรแกรมย่อยตามที่ต้องการแล้ว สั่ง RUN
พลอตเตอร์จะเริ่มทำงาน

5) เมื่อโปรแกรมนี้ทำงานถูกต้องตามที่ต้องการผู้ใช้สามารถเก็บโปรแกรมนี้ลงในแผ่นบันทึกได้ โดยใช้คำสั่ง SAVE "filename"

OK

SAVE "SINE WAVE"

OK

4.3 คำอธิบายประกอบการใช้โปรแกรม

ชุดของโปรแกรมย่อยนี้ประกอบด้วยโปรแกรมย่อย 10 โปรแกรม ซึ่งมีเลขกำกับคำสั่ง ตั้งแต่ 10000 ถึง 21040 ดังนั้น เลขกำกับคำสั่งดังกล่าว ผู้ทำโปรแกรมจะนำไปใช้ไม่ได้ เลขกำกับคำสั่งที่ควรใช้คือ 1 ถึง 9999

1) โปรแกรมย่อย INIT

- หน้าที่

เตรียมพร้อมสำหรับการรับส่งข้อมูลและสั่งให้พลอตเตอร์เตรียมพร้อมโดยเลื่อนปากกาไปยังจุดเริ่มต้น

- รูปแบบ

GOSUB 10000

- พารามิเตอร์

ไม่มี

- หมายเหตุ

ก่อนการใช้พลอตเตอร์ต้อง เรียกโปรแกรมนี้และโปรแกรมนี้ต้องอยู่ในหน่วย

ความจำตลอดเวลา

2) โปรแกรมย่อย FACTOR

- หน้าที่

กำหนดค่าความยาวของแกน ค่าตัวเลขสูงสุดและต่ำสุดบนแกน

- รูปแบบ

X. LENGTH = a: Y. LENGTH = b: X. MIN = c: X. MAX = d

Y. MIN = e: Y. MAX = f: GOSUB 11000

- พารามิเตอร์

a <numeric constant or numeric variable name>

แสดงความยาวของแกน X หน่วยเป็นนิ้วตั้งแต่ 1 ถึง 14 โดยมีค่า

default เป็น 14



b <numeric constant or numeric variable name>

แสดงความยาวของแกน Y หน่วยเป็นนิ้วตั้งแต่ 1 ถึง 8 โดยมีค่า

default เป็น 8

c <numeric constant or numeric variable name>

แสดงค่าต่ำสุดบนแกน X

d <numeric constant or numeric variable name>

แสดงค่าสูงที่สุดบนแกน X

e <numeric constant or numeric variable name>

แสดงค่าต่ำสุดบนแกน Y

f <numeric constant or numeric variable name>

แสดงค่าสูงที่สุดบนแกน Y

-หมายเหตุ

$c < d$ และ $e < f$

ต้องเรียกโปรแกรมนี้ก่อนโปรแกรม AXIS และต้องอยู่ในหน่วยความจำ

ตลอดเวลา

- ตัวอย่าง

```
10 X.LENGTH = 10 Y.LENGTH = 8
```

```
20 X.MIN = 0: X.MAX = 3.1416: Y.MIN = -1: Y.MAX = 1
```

```
30 GOSUB 11000
```

3) โปรแกรมย่อย AXIS

- หน้าที่

เขียนแกน X แกน Y ฮีตบนแกน ตัวเลขบนแกน ข้อความบนแกน และข้อ

ความประกอบ

- รูปแบบ

```
X.TICK = a: Y.TICK = b: X.TITLE$ = c: Y.TITLE$ = d
```

```
M.TITLE$ = e: GOSUB 12000
```

- พารามิเตอร์
 - a <numeric constant or numeric variable name>
แสดงค่าระยะ ระหว่างขีดบนแกน X
 - b <numeric constant or numeric variable name>
แสดงค่าระยะระหว่างขีดบนแกน Y
 - c <string constant or string variable name>
แสดงข้อความที่จะเขียนบนแกน X ไม่เกิน 12 ตัวอักษร
 - d <string constant or string variable name>
แสดงข้อความที่จะเขียนบนแกน Y ไม่เกิน 12 ตัวอักษร
 - e <string constant or string variable name>
แสดงข้อความที่จะเขียนประกอบรูปภาพ

- หมายเหตุ

ต้องเรียกโปรแกรม FACTOR ก่อนโปรแกรมนี้

- ตัวอย่าง

```
10 X.TICK = 5: Y.TICK = 2
20 X.TITLE$ = "YEAR": Y.TITLE$ = "DOLLARS"
30 M.TITLE$ = "PERSONAL BUDGET"
40 GOSUB 12000
```

4) โปรแกรมย่อย LOCATE

- หน้าที่

ยกปากกาไปวาง ณ จุดที่เริ่มลากเส้นกราฟ

- รูปแบบ

```
X.VALUE = a: Y.VALUE = b: GOSUB 14000
```

- พารามิเตอร์

a <numeric constant or numeric variable name>
แสดงตำแหน่งของจุดตามแนวแกน X

b <numeric constant or numeric variable name>

แสดงตำแหน่งของจุดตามแนวแกน Y

- หมายเหตุ

a และ b มีค่าไม่เกินค่าสูงสุดและต่ำสุดตามที่ระบุไว้ในโปรแกรม FACTOR

- ตัวอย่าง

10 X.VALUE = 0: Y.VALUE = 0: GOSUB 14000

5) โปรแกรมย่อย PLOTA

- รูปแบบ

X.VALUE = a: Y.VALUE = b: GOSUB 15000

- พารามิเตอร์

a <numeric constant or numeric variable name>

แสดงตำแหน่งของจุดตามแนวแกน X

b <numeric constant or numeric variable name>

แสดงตำแหน่งของจุดตามแนวแกน Y

- หมายเหตุ

a และ b มีค่าไม่เกินค่าสูงสุดและต่ำสุดตามที่ระบุไว้ในโปรแกรม FACTOR

- ตัวอย่าง

10 X = 0

20 FOR X = 0 TO 6.28 STEP 0.01

30 Y = COS(X)

40 X.VALUE = X: Y.VALUE = Y: GOSUB 15000

50 NEXT X

6) โปรแกรมย่อย PLOTR

- หน้าที่

ลากเส้นจากจุดเดิมไปยังจุดใด ๆ

- รูปแบบ

X.VALUE = a: Y.VALUE = b: GOSUB 16000

- พารามิเตอร์

a <numeric constant or numeric variable name>

แสดงระยะห่างจากจุดเดิมตามแนวแกน X

b <numeric constant or numeric variable name>

แสดงระยะห่างจากจุดเดิมตามแนวแกน Y

- หมายเหตุ

a และ b มีค่าสอดคล้องกับค่าสูงสุดและต่ำสุดตามที่ระบุไว้ในโปรแกรม

FACTOR

- ตัวอย่าง

10 X.VALUE = 1: Y.VALUE = 1: GOSUB 16000

20 X.VALUE = 0: Y.VALUE = 1: GOSUB 16000

30 X.VALUE = 1: Y.VALUE = 0: GOSUB 16000

40 X.VALUE = 0: Y.VALUE = -1: GOSUB 16000

50 X.VALUE = -1: Y.VALUE = 0: GOSUB 16000

7) โปรแกรมย่อย CHAR

- หน้าที่

เขียนตัวอักษร ณ ตำแหน่งใด ๆ

- รูปแบบ

X.VALUE = a: Y.VALUE = b: DIRECTION = c: SIZE = d

CSTRING\$ = e: GOSUB 17000

- พารามิเตอร์

a <numeric constant or numeric variable name>

แสดงตำแหน่งที่เริ่มเขียนตัวอักษรตามแนวแกน X

b <numeric constant or numeric variable name>

แสดงตำแหน่งที่เริ่มเขียนตัวอักษรตามแนวแกน Y

c <numeric constant or numeric variable name>

แสดงทิศทางของการเขียนตัวอักษร โดย 1 หมายถึง เขียนตามแนวนอน

และ 2 หมายถึง เขียนตามแนวตั้งจากบนลงล่าง

d <numeric constant or numeric variable name>

แสดงขนาดของตัวอักษรที่เขียนโดย

1 หมายถึง ขนาด 14/100 นิ้ว

2 หมายถึง ขนาด 24/100 นิ้ว

3 หมายถึง ขนาด 7/10 นิ้ว

4 หมายถึง ขนาด 1.4 นิ้ว

5 หมายถึง ขนาด 2.8 นิ้ว

e <string constant or string variable name>

แสดงชุดของตัวอักษรที่ต้องการเขียน

- หมายถึง

ตำแหน่งของการเขียนตัวอักษรเป็นตำแหน่งจริงบนแกนหน่วยเป็นนิ้ว

- ตัวอย่าง

10 X.VALUE = 1: Y.VALUE = 1: DIRECTION = 1: SIZE = 3

20 CSTRING\$ = "***HI I'M A PLOTTER**" : GOSUB 17000

8) โปรแกรมย่อย CIRCLE

- หน้าที่

เขียนวงกลม ณ ตำแหน่งใด ๆ

- รูปแบบ

X.VALUE = a: Y.VALUE = b: RADIUS = c: GOSUB 18000

- พารามิเตอร์

a <numeric constant or numeric variable name>

แสดงตำแหน่งของจุดศูนย์กลางของวงกลมตามแนวแกน X

b <numeric constant or numeric variable name>

แสดงตำแหน่งของจุดศูนย์กลางของวงกลมตามแนวแกน Y

c <numeric constant or numeric variable name>

แสดงความยาวของรัศมีของวงกลมขนาดไม่เกิน 5 นิ้ว

- หมายเหตุ

ตำแหน่งของจุดศูนย์กลางของวงกลมเป็นตำแหน่งจริงบนแกนหน่วยเป็นนิ้ว และรัศมีเป็นความยาวจริงมีหน่วยเป็นนิ้ว

- ตัวอย่าง

10 X.VALUE = 4: Y.VALUE = 4: RADIUS = 3: GOSUB 18000

9) โปรแกรมย่อย QUIT

- หน้าที่

ยกปากกากลับไปจุดเริ่มต้น

- รูปแบบ

GOSUB 19000

- พารามิเตอร์

ไม่มี

- หมายเหตุ

โปรแกรมนี้ใช้เมื่อการพลอตกราฟเสร็จสิ้นและต้องการเลิกใช้พลอตเตอร์

10) โปรแกรมย่อย CPCOM

- หน้าที่

ส่งคำสั่งที่ถูกต้องตามรูปแบบของพลอตเตอร์ซีพี-01 ไปยังพลอตเตอร์

- รูปแบบ

GOSUB 20000 'PU

GOSUB 20100 'MO

GOSUB 20200 'PD

GOSUB 20300 'ML

GOSUB 20400 'MR

GOSUB 20500 'MU

GOSUB 20600 'MD

GOSUB 20700 'PA

GOSUB 20800 'PR

GOSUB 20900 'CH

GOSUB 21000 'CV

- พารามิเตอร์

ไม่มี

- หมายเหตุ

โปรแกรมนี้ใช้เมื่อต้องการเปลี่ยนแปลงหรือพัฒนาโปรแกรมภาษา เบสิกใน

ขั้นต่อไป