

怎样给 TI59 手持式 计算机编程序

潘正伯 编著

——山东矿业学院——

序

电子计算机小型化为80年代新兴科学及事业，而习于大机者，辄以为不足道。谚云：“麻雀虽小，五脏俱全”，殊不知正以其小巧袖珍，而益臻其善也。目前小型可编程序袖珍机品类繁多，不下数十种，其负盛誉者：HP—41C/41CV、TI59/58C，这些计算机相当于1962年的一占地数房间的整套设备，于今缩小于掌握之中，其功能能判断、能存储、能循环、能编程、能打印、能记录，几乎与大型机无异，其差别仅在容量而已。

对于近代科学需大容量机者固属殷切，而着眼于离散算题，无求于许多容量者亦更为迫切。

特别对于数学中之特殊函数、无穷级数求和、绝大部分常用数值积分、概率与统计，以至利用电子计算机辅助设计等诸方面，则边设计边计算，有此一机在手，可专心致志于考虑设计等诸问题，而繁杂纷纭之计算可委诸袖珍计算机，昔日梦寐以求者竟成事实，其便利学者，嘉惠士林，实非浅鲜。

潘正伯老师，好学深思，学多精进，近将其编程心得，公诸学人，则习AOS程序体系者，其艰涩之处，了然在目，不致问难无门，望机兴叹。拥有AOS编程之袖珍计算机者，可发挥既有设备的效能，节约四化工作中之人力、时间，加速计算工作机械化自动化，诚大事与幸事也，故乐为之序。

李学荣 1982.3.12夜

序于北京铁道研究院宿舍

前 言

1958年首创集成电路、1963年发明TTL集成电路的美国德州仪器公司(Texas Instruments)从七十年代中期起,陆续推出TI57、TI58、TI58C和TI59等几种型号的可编程序计算器。TI59是其中内存最大、功能最强的一种。

这种计算器能够提供类似电子计算机的功能,而且体积小(连提包在内 $180 \times 100 \times 40$)、重量轻、携带方便,对环境条件无特殊要求,既有磁卡提供外存之便,又有PC-100C热印机与它配合,作输出设备使用。去年二月美国出版的计算机杂志(SIMULATION)上已经将它归入“手持式计算机”(Hand-Held Computer)一类。

这种计算器进入我国市场后,也博得了我国科技界的好评。认为它是一种价廉物美比较理想的超小型计算工具。

袖珍计算器在我国各管理部门和科技界,已被普遍采用,而可编程序计算器的引进和采用,只能说是刚刚开始。我们不妨设身处地的想一想:当越来越多的人第一次坐到这种计算器前时,他们最迫切的要求是什么?在我国现有条件下,购买并使用TI59的用户们,他们的需要和兴趣肯定说是集中于“可编程序”这四个字上的,他们最迫切的要求是学会驱使计算器尽快尽好地进行程序运算。重点阐明如何编程序,这是本书的第一个出发点。

近十年来BASIC、ALGOL、FORTRAN等几种常用高级语言已经在我国科技界逐步得到普及。我们假定在TI59的用户中有一部分是熟悉某些高级语言的,另有一部分现在虽不熟悉但今后总要去学的。而高级语言的确比TI59的汇编语言更接近于人类思维的习惯和数学演算的顺序。因此本书的第二个出发点是力求在两者之间建立某些联系,使学过高级语言的人,很快就能掌握TI59的汇编语言;不懂得高级语言的人,在掌握TI59的汇编语言之后,转而学高级语言时也会收到触类旁通的功效。

“不怕不识货,就怕货比货”,多年来流传于民间的这句谚语,指出了认识事物的一条重要途径。本书尽可能多配搭一些例题,而且将某些例题反复修改、逐步优化。使初学者能循着这些例题,层层加深对TI59程序设计的认识,提高自己设计和阅读程序的能力。

近年来“题海战术”遭到舆论的指责,一蹶不振。程序设计是一种技艺,就像跳水和体操一样。要想很好地掌握一种技巧,除了大量地刻苦地练习之外我们不知道还有甚么更好的更省劲的办法。Donald M. Monro在他的《Fortran实用教程》(Computer with FORTRAN A practical Course, 1977)中要求学生在学习该书第二到五章时,每章至少要有10小时上机时间,并尽可能多作题。本书作者认为这个意见是非常宝贵的。因此我们在每章之后都附了经过严格挑选的一批习题,建议读者以 $1/5$ 的时间听讲或自己看书,拿出 $4/5$ 甚至更多的时间去写程序和改程序。如果你认真这样办了,将来你会体会到:这才是

真正的“捷径”。这是本书的第四个出发点。

此书编写过程中得到山东矿业学院科研处、教务处、建井系、采煤系、计算机室和情报科领导和同志们热情指导和帮助，北京铁道研究院李学荣高级工程师为本书写了序言。在此谨表示衷心的感谢！

由于作者使用这种计算器的时间不长，知识浅薄，书中错误难免，如蒙发现，恳请指正。

编著者

81年12月于泰安

82年3月改于北京

目 录

序.....	i
前言.....	iii
第一章 键盘功能.....	1—17
1—1 电源开关.....	1
1—2 清除键.....	1
1—3 数据输入键.....	2
1—4 基本运算符.....	2
1—5 括符.....	3
1—6 第二功能键.....	4
1—7 代数运算键.....	4
1—8 分度制的选择.....	5
1—9 三角函数.....	5
1—10 反函数键.....	6
1—11 存入、读出键.....	7
1—12 存储器直接运算.....	7
1—13 转换键.....	8
1—14 数据变换键.....	10
1—15 显示控制.....	10
1—16 基本统计运算.....	11
1—17 回归、相关和趋势分析.....	13
练习.....	15
第二章 怎样编程序(上)——基本概念.....	18—60
2—1 什么是程序?.....	18
2—2 程序设计要点.....	18
2—3 流程图的意义.....	19
2—4 流程图的绘制.....	20

2—5	TI59的指令系统	25
2—6	程序编辑键	26
2—7	程序设计概貌	27
2—8	几种常用语句	29
2—9	语法错误	48
2—10	程序举例	49
	练习	56
第三章 主程序库的应用		61-84
3—1	主程序库简介	61
3—2	库程序的使用指令	62
3—3	主程序库应用举例	78
3—4	主程序库的有关资料	82
	练习	83
第四章 怎样编程序(下)——发展技巧		85-128
4—1	程序的书写格式	85
4—2	子程序	86
4—3	特征位	89
4—4	间接指令系统	92
4—5	特殊操作键	96
4—6	库程序选析	97
4—7	用库程序作子程序	115
4—8	程序的优化	117
	练习	123
第五章 磁卡和热印机的使用		129-136
5—1	磁卡	129
5—2	录写磁卡	129
5—3	磁卡的读入	130
5—4	热印机	130
5—5	TI59的输出码和输出格式	131
5—6	印刷符号对照表	135
后记		136
参考书目		137

第一章 键盘功能

如果我们把T159的可编程序、自动运算看作一种高级功能，那末，它同时还具备像普通计算器一样按键操作、手动运算的低级功能。这两种功能是兼容的。也就是说在程序输入计算器后，仍可作手动运算，而程序不被破坏。本章将对各个按键的功能分组加以介绍。

1—1 电源开关

T159采用小型贮电池，其电源开关设于计算器顶端，开关按钮向右推为“开”，向左推为“关”。电源一旦关掉，内存的全部数据和程序均清零。

1—2 清除键

CLR总清——把显示的数据和整个操作步骤清除掉，使显示寄存器（x寄存器）中的数字变为0。

CE输入清除——在运算符之后接着总是输入数据，如果发现输入数据出错，可用此键清除，而不破坏这个运算符和它前面的输入数据。

〔例〕原拟输入 $3 + 2 - 5 = 0$ 操作时误为

$3 + 2 - 3$ 可以按 **CE**，改过来，即

$3 + 2 - 3$ **CE** $5 = 0$

使用**CE**键对悬而未决的运算无影响，且不能清除 π 、T寄存器、计算结果和从数据存储器中调出的数据。使用**CLR**键则将一切悬而未决的运算均清除，但它不能清除数据和程序存储器、T寄存器、角度制的选择和小数点后位数的选择。

当运算出错或操作错误时，显示器中的数字跳动闪光。此时按**CLR**和**CE**都能停止闪光，但**CE**不破坏运算步骤，**CLR**则将全部运算步骤清除。

CP程序清除——手动时它能清除全部地址和程序存储器，子程序返回寄存器，T寄存器，复原特征位并将程序恢复到000步。如在程序中编进**CP**，则只起清除T寄存器的作用。

CMs数据存储器总清——除了X寄存器、T寄存器和程序存储器外，将数据存储器全部清零。

1—3 数据输入键

[0]—[9] 输入 0 至 9 任何数据。

[.] 输入小数点——此键以第一次按动为准，同一个数据中重复按动此键不起作用。

〔例〕如果按下 **[1][9][.][8][5][.][9][7][3]**

显示器中只出现 19.85973 后面那个小数点被拒入。

[+/-] 正负变换——显示器中为正值时，按此键将它变为负；反之，变为正。一般应在数据输入后再输入符号，否则容易出错。

[π] 输入 π ——此键可向显示寄存器输入13位 π 值 (3.141592653590)，显示器中可以看到前十位 (3.141592654)。

任意数值输入前不必使用清除键，显示器中已经使用过的数据会自动消失。

〔例〕在计算 $158 \div 2.5 = 63.2$ 之后紧接着计算

$$69.3 \times 17 = 1178.1$$

按键顺序是：**[1][5][8][+][2][.][5][=]** 62.2 (显示)，

[6][9][.][3][$\times$][1][7][=] 1178.1 (显示)

这给连续运算节省了许多清除操作。

输入不带整数部分的纯小数时，可以只按 **[.]** 开始，计算器自动在小数点前添 0。

〔例〕0.9533，可以只按 **[.][9][5][3][3]** 结果却显示 0.9533

1—4 基本运算符

[+]、**[-]**、**[$\times$]**、**[$\div$]** 和 **[=]** 这些符号的使用和书写顺序与算术或代数的规定一致，叫做 AOS 制 (代数运算制)。也就是说，计算器自动按照下列优先级的顺序进行运算：

1) 标准函数如 $\sin$ 、 $\cos$ 、 $\ln x$ 、 $\log$ 以及它们的反函数、直角坐标与极坐标的互换、度分秒的互换、 x^2 以及 $\sqrt{x}$ 等，只要按键下去，立刻显示出来。

2) 其次是幂和方根 (y^x 和 $\sqrt[x]{y}$)。

3) 然后是乘、除。

4) 最后是加、减。

同一级中按输入的先后次序运算，这就是 AOS 制。

〔例〕求 $3 + 0.25^5 - \sin 30^\circ \times 14 + 17 \div 18 = ?$

执行的顺序如下：

$$3 + 0.25^{y^x} 5 - 30 \sin \times 14 + 17 \div 18$$

② 0.0009765625 ① 0.5

⑤ 3.000976563 ⑧ 7 ④ 9444444444

⑥ -3.999023438

⑦ -3.054578993

$$\therefore 3 + 0.25^5 - \sin 30^\circ \times 14 + 17 \div 18 = -3.054578993$$

要想改变运算顺序，就得运用括号（〔和〕）。

1—5 括号

〔前括号，〕后括号——用法与数学上的习惯一致。

TI59允许括号套括号达9重之多。每个后括号按下去，计算器将该层括号内的运算结清并显示出来。

〔例〕 $(2 + 1)(3 + 2) = ?$

按键顺序〔2〕〔+〕〔1〕〔〕〔×〕〔〔〕3〕〔+〕〔2〕〔〕〕〔=〕

结果显示15。注意，两对括号中必需有个×，否则运算就无法进行下去。

〔例〕求解 $\frac{8 \times (4 + 9) + 1}{(3 + 6 \div 2) \times 7} = ?$

说明：以下对按键操作，只写符号不画边框。

按 键	显 示	注 解
CLR	0	总清
(8 × (4 + 9)	13	求出 4 + 9
+	104	求出 8 × (4 + 9)
1)	105	求出分子
÷ ((3 + 6 ÷ 2)	6	求出 (3 + 6 ÷ 2)
× 7)	42	求出分母
=	2.5	得出答案

〔CE〕键与括号配合使用时，可以将显示值带进括号，从而节省了输入数值的手续和时间。

〔例〕 $3.296214 + (3.296214 \times 6) = 23.073498$

按 键	显 示	注 解
CLR 3.296214 +	3.296214	
(CE ×	3.296214	CE将3.296214带入括号
6)	19.777284	
=	23.073498	

1—6 第二功能键

[2nd]第二功能键——观察TI59的表盘可以看出，每个按键上印有白色或黑色的文字或符号，这就是我们说的第一功能。再看许多按键上方的表盘上还印有金色的文字和符号，这就是我们说的第二功能。例如第二行第三列这枚按键的第一功能是 $\ln x$ ，它的第二功能是 $\log$ ；再如第八行第四列这枚按键的第一功能是数码3，它的第二功能是数值 π 。

按动按键，就产生第一功能的作用；先按[2nd]，再按某个按键，就产生该按键的第二功能的作用。例如按下[2nd][x^2]就等于按下[sin]键一样。为了使程序清楚，删去冗枝，本书中一律将2nd省去，直接写出它的第二功能。例如作上述操作时，本书直接写sin，而不写2nd x^2 或2nd sin字样。用户在使用时希望留心，不要忘了按[2nd]，以致第把一功能与第二功能用混了。

1—7 代数运算键

[$1/x$]倒数——将显示值 x ，变为它的倒数 $1/x$

〔例〕求3.2的倒数

按 键	显 示
3.2 $1/x$	0.3125

如果显示为0，按[$1/x$]将导致显示值的跳动闪光。

[x^2]平方，[$\sqrt{x}$]开平方——求显示值的平方或它的平方根。

〔例〕求 $(3.1+4.3)^2$

按 键	显 示
$(3.1+4.3) x^2$	54.76

〔例〕求 $\sqrt{0.76-0.16}$

按 键	显 示
$(.79+.16) \sqrt{x}$	0.7937253933

注意此例中如果不加括符按[$\sqrt{x}$]后显示0.4，只是将0.16开平方。

负数开平方导致显示值的闪光。

[lnx]自然对数，[log]常用对数——分别给出以 e ($=2.718281828$)为底的或以10为底的对数值。

〔例〕 $\log(1+\ln 1.7)=?$

按 键	显 示
CLR	0
(1 +	1.
1.7lnx)	1.530628251
log	.1848697249

[INV] [lnx]自然反对数——给出自然反对数 e^x 之值；其工作范围为 $-227.9559242 \leq x \leq$

230.2585092。超出此范围导致显示值的闪光。

INV log 常用反对数——给出常用反对数 10^x 之值，其工作范围为 $-99 \leq x \leq 99.99999$ 998，超出此范围导致闪光。

〔例〕 $e^{(3+10^{0.3})}=?$

按 键	显 示
CLR (3 +	3
. 3 INV log	4.995262315
INV ln x	147.7116873

y^x 乘x次幂——计算y的x次幂。其按键顺序为y^x然后按运算键或等号时显示结果。

〔例〕求 $3.72^{0.75}=?$

按 键	显 示
CLR 3.72	3.72
y ^x .75	0.75
=	2.678595932

INV y^x 开x次方——计算y的x次方根。其按键顺序为y^{INV y^x}然后按运算键或等号时显示结果。当y<0时，或y=0而x<0时导致闪光。

〔例〕求 $\sqrt[3]{2.36^{-0.23}}=?$

按 键	显 示	注 解
(2.36 y ^x	2.36	输入y及y ^x
.23 +/-	-0.23	输入x
) INV y ^x	.8207865654	输出 $2.36^{-0.23}$
3 =	.9362893421	输出最终结果

1—8 分度制的选择

Deg 角度制——即通常使用的将一周分为 360° 的制度 但需留意，使用Deg时，角度之后不是分秒，而要将分秒化为度的小数形式。即 $17^\circ 15'$ 应化为 17.25° 参加运算。

计算器从OFF进入ON（开机后重新开机），如果用户不作分度制的选择，计算器将自动选择角度制。

Rad 弧度制——即数学上常用的将一周分为 2π 弧度的制度。

Grad 百分度制——这是另一种分度的方法，它将一周分为400度。

分度制选择按键在运算中只按一次，只要不关机、不另选分度制，它将始终起作用，CLR、CE都不能清除它。

1—9 三角函数

Sin 正弦，**Cos 余弦**，**tan 正切函数键**——分别给出显示值的正弦值、余弦值或正切值。

将这三个键与倒数键相配合，又能产生另外三个三角函数：

$$\boxed{\text{Sin}} \boxed{1/x} = \text{CSC 余割}$$

$$\boxed{\text{Cos}} \boxed{1/x} = \text{SEC 正割}$$

$$\boxed{\text{tan}} \boxed{1/x} = \text{COT 余切}$$

在三角函数之前加上 $\boxed{\text{INV}}$ 就可以得到它的反三角函数。其余三个反三角函数也可以求得

$$\boxed{1/x} \boxed{\text{INV}} \boxed{\text{Sin}} = \text{arccsc}$$

$$\boxed{1/x} \boxed{\text{INV}} \boxed{\text{Cos}} = \text{arcsec}$$

$$\boxed{1/x} \boxed{\text{INV}} \boxed{\text{tan}} = \text{arccot}$$

1—10 反函数键

$\boxed{\text{INV}}$ 反函数键——求某些函数的反函数或某些运算的逆运算。在按了 $\boxed{\text{INV}}$ 之后，函数或运算都产生如下的变换：

函数或运算	反函数或逆运算
$\ln x$	e^x
$\log$	10^x
y^x	$x\sqrt[y]{y}$
$\sin$	arc sin
$\cos$	arc cos
$\tan$	arc tan
$P \rightarrow R$ (极坐标变直角坐标)	$R \rightarrow P$ (直角坐标变极坐标)
$\Sigma +$ (累加)	$\Sigma -$ (累减)
EE (以指数形式显示)	取消EE (按常规显示)
ENG (以工程记数法显示)	取消ENG (按常规显示)
Fix (固定小数点后留几位)	取消Fix (按全字长显示)
Int (取整数)	取小数
D.MS (度.分秒)	化度分秒为10进制度数
Prd (在存储器内作乘法)	在存储器内作除法
SUM (在存储器内作加法)	在存储器内作减法
SBR (子程序)	返回主程序
$x=t$	$x \neq t$
$x \geq t$	$x < t$
ifflg (若有特征位)	若无特征位
stflg (建立特征位)	撤消特征位
Dsz (逢零跳一步)	非零跳一步
write (写上)	读入
list (列表印出程序)	列出存储器中的数据

当INV与2nd共同使用时，在手动运算时，INV与2nd之间的先后次序对运算结果不产生影响，但在自动运算时，要求将INV排在2nd之前。

1—11 存入、读出键

STO $\times\times$ 存入——将显示值存入 $\times\times$ 号存储器，（以后我们用 $R_{\times\times}$ 表示第 $\times\times$ 号存储器）一个新的值存入 $\times\times$ 时，以前存的数据被清除；此新值一直保存到有更新的值存入才被清除。

RCL $\times\times$ 读出——将 $\times\times$ 号存储器的内容读出。显示器上立刻显示此值，但 $\times\times$ 号的内容依然不变。

〔例〕将3.012存入23号存储器（ R_{23} ），然后再将它读出。

按 键	显 示	注 解
3.012 STO 23	3.012	将3.012存入 R_{23}
CLR	0	将显示器清零
RCL 23	3.012	将 R_{23} 的内容读出

利用存储器可以节省许多重复性动作，即能节省时间，又能减少差错。

〔例〕计算 $4x^3 + 3x^2 - x - 7.1$ 之值，此处 $x = 2.94672813$

按 键	显 示	注 解
CLR $4 \times$	4	
2.94672813 STO 10	2.94672813	将2.94672813存入 R_{10}
$y^x 3 +$	102.3481974	算出 $4x^3$
$3 \times RCL 10 x^2$	8.683206672	算出 x^2
$- RCL 10$	2.94672813	读出 x
$- 7.1$	7.1	输入常数项
$=$	118.3510893	最后结果

此题如不用存储键，就要反复送三次2.94672813，非常容易出错。

1—12 存储器直接运算

在此以前，一切运算都是在 x 寄存器中进行的，如果采用下列按键，运算将直接在存储器内进行，运算结果也不在 x 寄存器中显现。

SUM $\times\times$ 存储器内求和——将显示寄存器的值加到存储器 $R_{\times\times}$ 中。

INV SUM $\times\times$ 存储器内求差——将存储器 $R_{\times\times}$ 的值减去显示寄存器的值，结果存放在 $R_{\times\times}$ 中， $R_{\times\times}$ 的原先值被清除。

Prd $\times\times$ 存储器内求积——将显示值乘以 $R_{\times\times}$ 值，其积存入 $R_{\times\times}$ ， $R_{\times\times}$ 的原先值被清除。

INV Prd $\times\times$ 存储器内求商——存储器 $R_{\times\times}$ 的值被显示值除，其商存入 $R_{\times\times}$ ， $R_{\times\times}$ 的原先值被清除。

〔例〕求 $\sum_{x=1}^3 (x^2 + 9)$ 之值。

按 键	显 示	03号存储器的内容
$1x^2+$	1	0
$9=STO\ 03$	10	10
$2x^2+$	4	10
$9=SUM\ 03$	13	23
$3x^2+$	9	23
$9=SUM\ 03$	18	41
RCL 03	41	41

〔例〕求 $(28+6.6) \times 1.05 = ?$

按 键	显 示
28 STO 01	28
6.6 Sum 01	6.6
1.05 Prd 01	1.05
RCL 01	36.33

注 解
将28存入R₀₁
将6.6加进R₀₁
将R₀₁乘1.05
将计算结果从R₀₁中读出

1—13 转换键

[D.MS]度分秒变换——度、分、秒变换也就是时、分、秒变换。它的功能是将显示寄存器中以度、分秒（或时·分秒）表示的数值变换成十进制的角度（或小时）数。

度、分、秒（或时、分、秒）的输入形式是将小数点置于度（时）之后，分占两位数字，其后为秒。

〔例〕 $17^\circ 5' 25''$ 输入时为17.0525

$183^\circ 47' 13.333''$ 输入时变成183.4713333

〔例〕将 $20^\circ 30'$ 转换为十进制度数

按 键	显 示
20.30	20.30
D.MS	20.5

注 解
将度分秒按规定格式输入
转换成了十进制度数

[INV][D.MS]十进制度数变换——即将十进制度（时）数，变成度（时）分秒数。

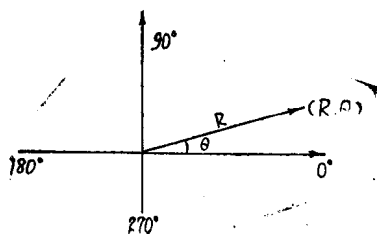
〔例〕将 31.7426° 换成度、分、秒形式。

按 键	显 示
31.7426	31.7426
INV[D.MS]	31.443336

注 解
输入十进制度数
换算成 $31^\circ 44' 33.36''$

[x $\rightleftharpoons$ t] **x,t变换键**——将显示寄存器（x）的值与检测寄存器（t）的值相交换。t寄存器是独立于全部存储器之外的。**[x $\rightleftharpoons$ t]**键是把显示数据送入t寄存器，同时t寄存器的内容引致显示寄存器的唯一通道。

[P $\rightarrow$ R] **极坐标变为直角坐标**，**[R $\rightarrow$ P]** **直角坐标变为极坐标**——请看下面两个例子



极坐标换直角坐标

- 1) 输入 R
- 2) 按 $x \Rightarrow t$
- 3) 输入 θ
- 4) 按 $P \rightarrow R$, 显示 y 值,
- 5) 按 $x \Rightarrow t$, 显示 x 值,

输入时先要选好 θ 的分度制。 θ 规定的上下界是:

$$\left. \begin{array}{l} -90^\circ \\ -\pi/2 \text{ 弧度} \\ -100 \text{ 百分度} \end{array} \right\} \leq \theta < \left\{ \begin{array}{l} 270^\circ \\ 3\pi/2 \text{ 弧度} \\ 300 \text{ 百分度} \end{array} \right.$$

〔例〕将下列极坐标值化为直角坐标值

$$R = 5, \theta = 30^\circ$$

选用角度制。

按 键	显 示	注 解
5 $x \Rightarrow t$	0	输入 R
30 $P \rightarrow R$	2.5	输入 θ , 显示 y 值
$x \Rightarrow t$	4.330127019	显示 x 值

〔例〕已知 $x = 3, y = 4$, 求它的以弧度表示的极坐标

按 键	显 示	注 解
CLR Rad	0	选择弧度制
3 $x \Rightarrow t$	0	输入 x 值
4 INV $P \rightarrow R$	0.927295218	输入 y 值, 显示 θ 的弧度值
$x \Rightarrow t$	5	显示 R 值

EXC $\times \times$ 存储交换——将 x 寄存器的内容与第 $\times \times$ 号存储器的内容交换。在检查存储器的内容时, 使用此键可同时检查两个存储器, 而不会把它们中的任一个丢失。你还可以把某些数据暂存起来, 以后再用。

〔例〕已知 $A = 0.258963, B = 1.255632$

求 $A^2 + AB + 2B^2$ 之值

按 键	显 示	注 解
.258963 STO 13	0.258963	将A存入R ₁₃
$x^2 + 1.255632 \times$	1.255632	输入B
Exc 13	0.258963	存入B, 调出A
+ 2 ×	2	
RCL 13	1.255632	调出B
$x^2 =$	3.545447504	答案

1—14 数据变换键

[X] 绝对值——对显示数值取绝对值。

[Int] 取整——将显示数值的小数部分舍去, 不管小数部分有多大。只保留它的整数部分。舍去的部分再也不能恢复。

〔例〕将-127.999999取整

按 键	显 示	注 解
123.99999 +/-	-127.99999	输入-127.99999
Int	-127	取整后符号不变

[INV] [Int] 舍整——将整数部分舍弃, 只保留符号和小数部分。

〔例〕将-127.99999舍整

按 键	显 示	注 解
127.99999 +/-	-127.99999	
INV Int	-0.99999	

1—15 显示控制

本机显示屏上允许出现10位有效数字, 另有一位小数点, 一位符号位, 为了保证这10位数字的准确性, 机器以13位数进行运算, 将其结果四舍五入以10位数字输出。如果整数部分超过10位数字, 机器将自动以科学记数法表示。

[EE] 输入指数——按此键后输入的是以10为底的指数, 而且以后显示的数字, 均以科学记数法表示, 直到CLR或关机为止。

〔例〕将32000000000输入x寄存器, 先将它变成 3.2×10^{11} 的形式

按 键	显 示	注 解
CLR	0	
3.2	3.2	
EE	3.2 00	
11	3.2 11	

本机可以处理的数字范围在 $\pm 1 \times 10^{-99}$ 到 $\pm 9.999999 \times 10^{99}$ 之间, 超过此范围产生溢出或下溢, 都将导致显示值闪光。

[INV] [EE] 消除科学记数法——使显示值变为普通形式。

ENG 工程记数法——工程上常按千单位折算（如千克、千米、千瓦、千伏安、千欧、毫伏、微伏、兆欧等），这是一种指数按 3 增减的科学记数法（如 10^3 、 10^6 等）。

〔例〕求圆周长为 3×10^{-3} 公尺时直径是多少微米（1 微米 = 10^{-6} 米）。

按 键	显 示
CLR Eng	0. 00
3 EE	3. 00
3 +/÷	3. -03
$\pi =$	954.92966 -06

INV Eng 消除工程记数法——而按 CLR 或 CE 都不能达到此目的。

Fix 定位键——按此键后输入从 0 到 8 的一个数，计算器就按照这个数字确定小数点后保留几位数，自动按四舍五入原则整理。

〔例〕 $2/3 = .666666667$

按 键	显 示	注
2 ÷	2	
3 =	.666666667	
Fix 5	0.66667	小数后保留 5 位
Fix 2	0.67	保留两位小数
Fix 0	1	不要小数
INV Fix	.666666667	

〔例〕 $1 \times 10^{-3} \div 2 = 0.0005$

按 键	显 示	注
1 EE	100	用科学记数法
3 +/÷	1. -03	输入被除数
2 =	5. -04	求出结果
Fix 2	5.00 -04	小数后保留 2 位
INV EE	0.00	返回普通记数法
Fix 3	0.001	将 0.0005 四舍五入
Fix 4	0.0005	完整显示
Fix 5	0.00050	多一位添 0 补足 5 位

Pause 暂停键——为了观察程序的运行情况或中间结果，可以应用 Pause 键。在程序中编入此键的地方，程序运行至此，暂停半秒将 x 寄存器中的值显示出来，接着继续运行。如果按下 Pause 键不松手，计算器就将 x 寄存器内的运算全部显示出来。

暂停键主要功能是帮助我们试调程序，程序调好后，可将显示不必要的中间结果的暂停键删去，以提高程序运行的效率。

1—16 基本统计运算

由于统计运算中要占用 t 寄存器和 1 至 6 号数据存储器，故在进行统计运算前，先要依次