INSTRUCTION MANUAL

Serial Number ____________

PROGRAMMER 926
OPERATORS
MANUAL

Tektronix, Inc. —●— P.O. Box 500 —●— Beaverton, Oregon 97005 —●— Phone 644-0161 —●— Cables: Tektronix

Manufactured by Calculator Products Division —●— 1089 Morse Avenue —●— Sunnyvale, California 94086

070-1253-00 771

WARRANTY

All TEKTRONIX instruments are warranted against defective materials and workmanship for one year. Any questions with respect to the warranty should be taken up with your TEKTRONIX Field Engineer or representative.

All requests for repairs and replacement parts should be directed to the TEKTRONIX Field Office or representative in your area. This will assure you the fastest possible service. Please include the instrument Type Number or Part Number and Serial Number with all requests for parts or service.

Specifications and price change privileges reserved.

U.S.A. and foreign TEKTRONIX products covered by U.S. and foreign patents and/or patents pending.

TABLE OF CONTENTS

SECTION I DESCRIPTION

SECTION II USING THE PROGRAMMER 926

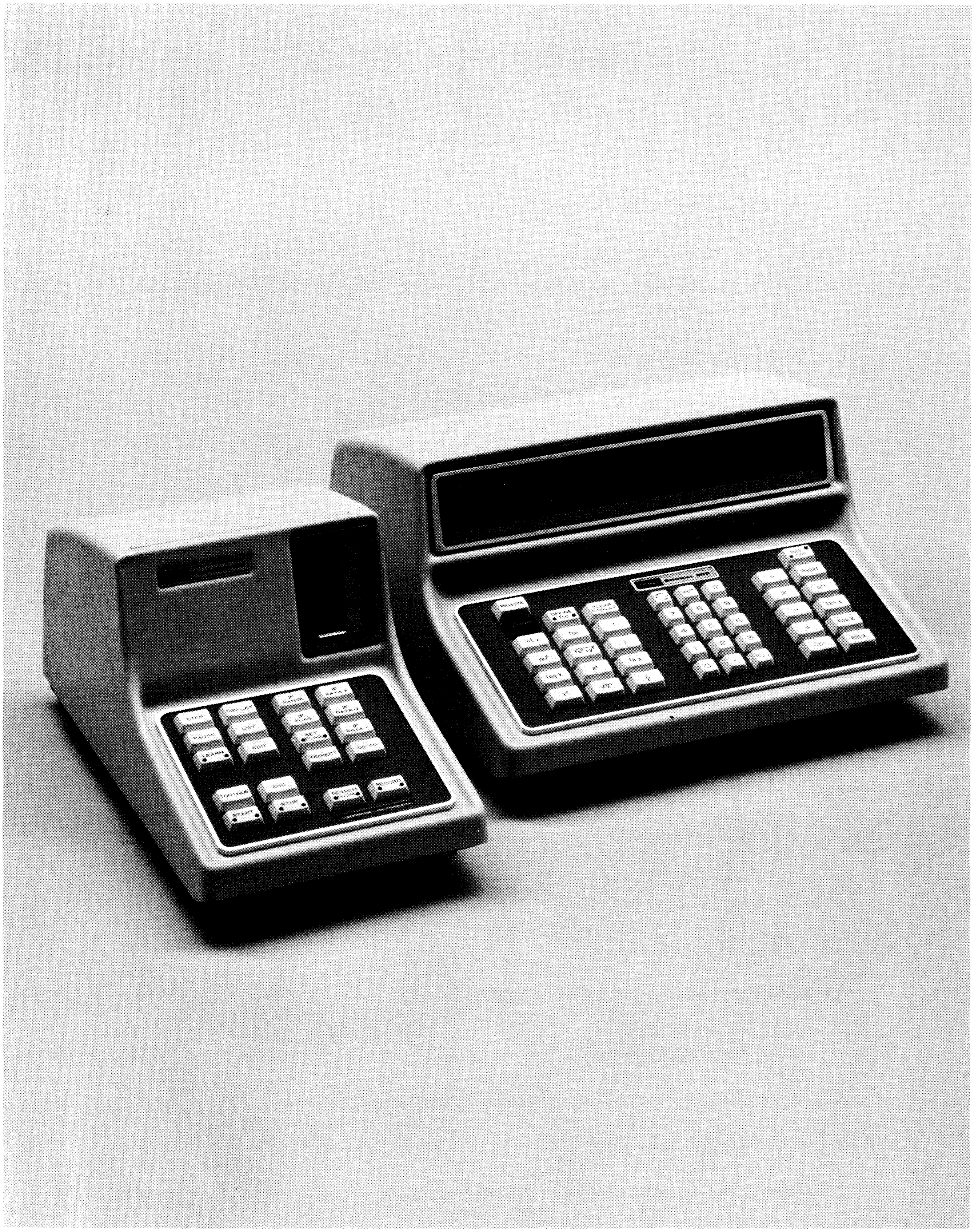


Fig. 1-1. Programmer 926 with Scientist 909 or Statistician 911 Calculator.

SECTION 1
DESCRIPTION

1.1 GENERAL DESCRIPTION

The Programmer 926 adds power to the basic Tektronix calculators. It extends the programming capability of the Scientist 909 or the Statistician 911 to permit computer-like decision making, branching, looping, subroutining and unlimited nesting of subroutines. Additionally, it provides permanent program storage on magnetic tape. Programs up to 512 steps long can be stored in the internal memory of the Programmer (each keystroke is a step). With the removable magnetic tape cartridge inserted in the Programmer, memory capacity is extended to 5120 steps, or 10 blocks of 512 steps. When the cartridge is used for program recording or playback, it interacts with the Programmer's internal memory.

Changing, deleting, and adding program steps are straightforward with the Programmer 926, since it has complete editing as well as programming capability. The contents of each program step and its step number can be visually displayed on the calculator before changes are made. Additionally, hard copy listings of all internally stored program steps may be printed using a peripheral Tektronix Printer 941.

1.2 MACHINE ORGANIZATION

The Programmer 926 is capable of being programmed and operating on programs.

1.2.1 Learn or Programming Mode

Indicated by lit LEARN key lights. When the Programmer 926 is learning, all keys behave as indicated by their titles. However, during program operation, functions of many of the Programmer keys are determined by the operating mode. Changing from one mode to another is accomplished by pressing certain keys, and accomplishment of the mode change is varified by key light indications.

1.2.2 Idle Operating Mode

Normally indicated by all key lights unlit. Randomly, SET FLAG or STOP may be lit, and may be extinguished by pressing IF FLAG xxx or END respectively.

1.2.3 Run Operating Mode

Signified by lit START key lights, LEARN and STOP key lights off.

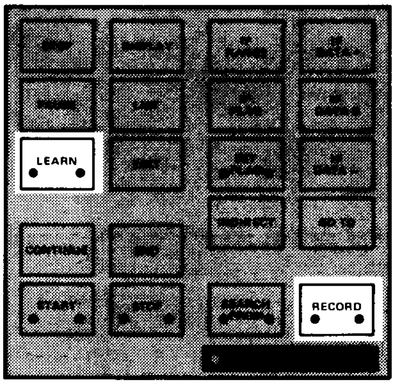
1.2.4 Stop Operating Mode

Signified by both START and STOP key lights being lit.

1.3 FUNCTIONS OF THE KEYS

The Programmer keys may be grouped into four functional categories: storage keys, unconditional keys, jump keys, and check/edit keys.

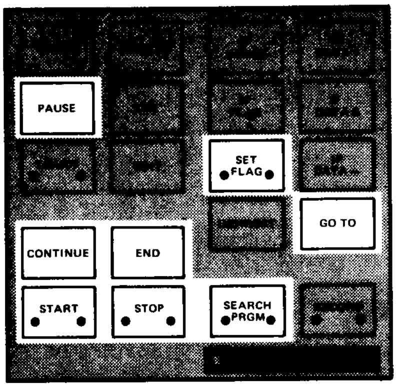
1.3.1 Storage Keys



These two non-programmable keys permit storage of program steps either in the Programmer's internal memory or on the magnetic tape cartridge. Storage of a program on tape is always from the internal memory and not directly from keystrokes. (Definition of non-programmable keys: keys not capable of being stored between pressings of LEARN.)

	IDLE	STOP	RUN
LEARN	Stores in internal memory programmable keystrokes made while the key lights are on. Storage begins at the step most recently accessed.		Has no effect (not programmable by Programmer).
RECORD	Followed by a one-digit address (0 to 9), causes the machine to access the specified program block on the tape cartridge (and the SEARCH PRGM lights to go on). Once the program is located (SEARCH PRGM lights go off), the program steps stored internally are transferred to the tape, beginning with the step in the counter when RECORD was pressed. While the actual transfer is taking place, all keys are inoperative.		Has no effect (not programmable by Programmer).

1.3.2 Unconditional Keys



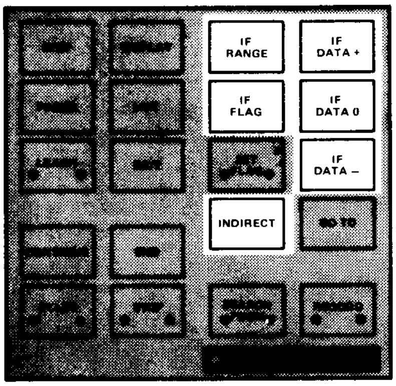
These eight keys unconditionally direct machine operation. Individual functions, as determined by the operating mode, are as follows:

	IDLE	STOP	RUN[1]
START	Begins program execution at step 000, provided STOP is off. If STOP is on, the machine jumps to 000 and goes into the STOP MODE.	Resets the step counter to 000.	Causes a jump to 000.
STOP	Has no effect (lights go on). If START is pressed afterwards, the machine jumps to 000 and goes into the STOP MODE.	Has no effect (lights remain on).	Halts execution of the program and puts the machine into the STOP MODE. If DISPLAY is pressed, the step counter indicates the address of the next step to be executed.
CONTINUE	Resumes execution of the program. This step is necessary when program execution begins at a step other than 000.		Acts as a dummy step.

[1] **While in RUN MODE, all key steps except STOP are encountered within a program rather than manually pressed.**

END	Causes a jump to 000.	Causes a jump to 000 and puts the machine into IDLE MODE (START and STOP lights go off).	Terminates a program run and causes a jump to 000.
SET FLAG	Sets the flag (key lights go on) so that the next IF FLAG xxx encountered causes a jump to step xxx, at which time the flag is reset (key lights go off). The flag may be set either by the user or by the program.		
SEARCH PRGM	Followed by a one-digit address (0 to 9) accesses the specified program block on the tape cartridge. The LEARN key lights come on. Then the program is transferred to internal storage, beginning with the step last accessed. When this is finished. the lights on LEARN and SEARCH PRGM go out, and execution of the program begins (START lights come on). Press LEARN then STOP to discontinue a program search. All keys are inoperative during the few seconds when actual transfer of a program takes place.		Has the same effect as when the machine is in IDLE or STOP MODE, except that the program is transferred into internal storage beginning with step 000.
GO TO	Followed by a three-digit address causes unconditional jumping to the location indicated.		
PAUSE	Halts execution for 1/2 second so that the display can be viewed. Two or more successive PAUSE steps lengthen display time in 1/2 second increments.		

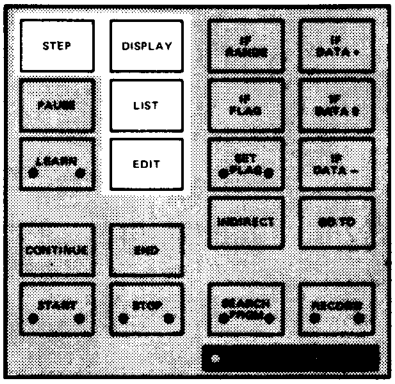
1.3.3 Jump Keys



These six keys command a conditional response, and are unaffected by the idle-stop-run operating mode conditions.

IF DATA +	Followed by a three-digit address, causes jumping if the data in the display is positive when the step is encountered. Note: A zero which is the result of a calculation is always positive.
IF DATA 0	Followed by a three-digit address, causes jumping if the display contains only zeros when the step is encountered.
IF DATA –	Followed by a three-digit address, causes jumping if the data in the display is negative when the step is encountered.
IF RANGE	Followed by a three-digit address, causes jumping when the display is flashing due to an answer which exceeds the allowed range, an imaginary root, etc.
IF FLAG	Followed by a three-digit address, causes jumping if the flag has been set (at which time the key lights on ●SET FLAG● go out).
INDIRECT	Causes a jump to the location indicated by the last three digits in the display when the step is reached. Thus, for example, the sequence K_{00} INDIRECT causes the machine to jump to the location designated by the last three digits of K_{00}.

1.3.4 Check/Edit Keys



These four non-programmable keys manipulate steps of a program, and are effective only while the Programmer is not running a program. The EDIT key is effective only in LEARN MODE.

	IDLE	STOP	RUN
DISPLAY	Displays the program step number, the key code and the block number for the step last accessed. Pressing STEP displays the same thing for successive program steps. Press DISPLAY a second time to restore the 10-digit number that was in the display before DISPLAY was pressed.		No effect
STEP	Advances the program to the next step.	Advances the program to the next step and causes it to be executed (providing the START lights are on). If DISPLAY is pressed, it must be pressed again before continuing.	No effect

LIST

IDLE	STOP	RUN
The sequence CLEAR 9 LIST prints out program steps automatically, beginning with the step last accessed. The printout consists of no line numbers and the same information displayed when DISPLAY is pressed. A listing may be terminated by holding STOP down. *NOTE* *9 is the address of the Tektronix Printer 941.*		No effect

LEARN MODE ONLY

EDIT

Each press of the key moves all steps of a program deeper into memory by one step, beginning with the step last accessed (example: step 50 moves to 51). The next programmable step pressed will be inserted at the location indicated on the counter. To insert more than one step, press EDIT before each new keystroke, or press EDIT the number of times required to enter all new keystrokes, then press the new keystrokes.

SECTION II

USING THE PROGRAMMER 926

To keep track of the sequence of program keystrokes stored in the Programmer, the keystrokes are numbered by sequential three-digit step numbers, 000 to 511. Step numbers are generated by an automatic step counter. The step number in the counter for the step last accessed can be viewed by pressing DISPLAY .

2.1 LEARNING

The basic difference between Programmer programs and calculator programs using DEFINE ●f(x)● is the sequence of keystrokes:

●LEARN● . . . program . . . ●LEARN●

which stores the program in the Programmer internal memory beginning at the step last accessed, rather than returning to the first step as is required by the calculator. Thus it is necessary to jump to a known location before storing program steps. Since this location is usually the first program step (000), the program step counter resets to 000 when program execution is completed or when END is pressed.

To learn a program, press END to go to step 000, or GO TO xxx (where xxx is the internal memory location to receive the first step of the program); then press ●LEARN● and the key lights turn on. All following programmable keystrokes of the Programmer, the Scientist 909 or the Statistician 911 are stored in the Programmer internal memory, and the step counter is incremented by one step for each keystroke. Furthermore, all keystrokes made on the calculator and ●SET FLAG● on the Programmer are executed. To terminate learning, if less than 512 steps are used, press ●LEARN● again and the key lights go out for verification. The ●LEARN● lights go out automatically when 512 program steps are reached.

2.2 CHECKING AND EDITING

The non-programmable DISPLAY, STEP, LIST, and EDIT keys fulfill checking and editing functions. Press DISPLAY to show, on the calculator, the step last accessed by the step counter, the key code (see APPENDIX A) of the step at that location, and the tape program block number (0 through 9) in which the program resides. If the program is not read from a tape, the block number display is meaningless.

Example display:

501	031	03
Program Step Number	Code For $\sqrt{x}$	Program Block Number

2.2.1 Changing Programs Without Adding Steps

To change a program, go to the step to be changed and learn the new step or series of steps as follows (using the previous example, to change $\sqrt{x}$ to x^2):

GO TO 501 LEARN x^2 LEARN

To check the program, press GO TO 501 DISPLAY and read: 502 032 03

Press DISPLAY again before continuing to restore the normal display.

2.2.2 Adding Program Steps

To insert program steps, such as K_{11} before step number 500, use the EDIT key as follows:

Press GO TO 500 LEARN EDIT K EDIT 1 EDIT 1 LEARN

This sequence of keystrokes moves all keystrokes beginning with step 500 back (deeper into memory) by the number of added keystrokes. In this example, the step which was number 500 before editing becomes 503, since three steps were added. All successive program steps are also renumbered by the three additional steps.

NOTE

When programming, jumps into this renumbered area of the program must be revised to show the new step number.

To view the coded information for the sequence K_{10} IF DATA + 296, where K is step number 205, press

GO TO	205	DISPLAY	STEP	STEP	STEP	STEP	STEP	STEP
Program contents:		K	1	0	IF DATA +	2	9	6

Next, press DISPLAY to restore the original 10-digit number to the display.

To execute this sequence, step by step, press:

GO TO 205 STOP START STEP STEP STEP STEP STEP STEP

Next, press DISPLAY to view the contents of step number 296, if data was +.

2.3 JUMPING

Programmer programs need not be written in continuous numerical sequence. Steps may be stored out of sequence and accessed with a jump to the step number of the first out-of-sequence step. Furthermore, part of a program can be stored for use in more than one place in an overall routine, in which case this part of the program becomes a subroutine.

There are four ways to jump from program step x to step y:

At x, command an unconditional jump to y. Key is GO TO .

At x, command that a jump to y be made only if the data displayed at that point fulfills the condition specified. Keys are IF DATA + , IF DATA 0 , IF DATA – and IF RANGE . In the latter case, "range" means that the dynamic range of the calculator is exceeded, or some other condition causes a flashing display.

At x, command a jump to y if a flag has been set in the program (or externally). Key is IF FLAG .

At x, command a jump to y, where y is the number in the display when the command is encountered. Key is INDIRECT .

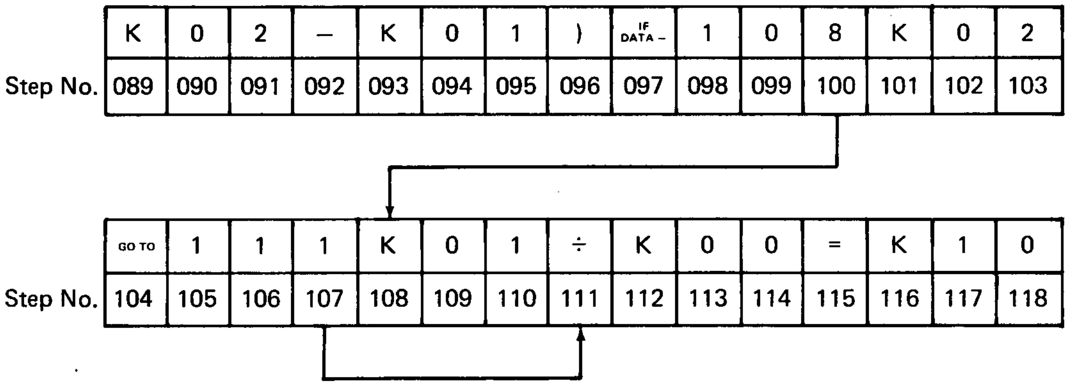
2.3.1 Jumping to the Middle of a Calculation

Jumps do not affect the straightforward keying sequence of the calculator. For example, for the calculation:

$$K_{01} \div K_{00} = K_{10} \qquad \text{if } K_{01} > K_{02}$$

$$K_{02} \div K_{00} = K_{10} \qquad \text{if } K_{01} < K_{02},$$

use the following programming steps to perform a conditional jump:



	K	0	2	–	K	0	1	)	IF DATA –	1	0	8	K	0	2
Step No.	089	090	091	092	093	094	095	096	097	098	099	100	101	102	103

	GO TO	1	1	1	K	0	1	÷	K	0	0	=	K	1	0
Step No.	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118

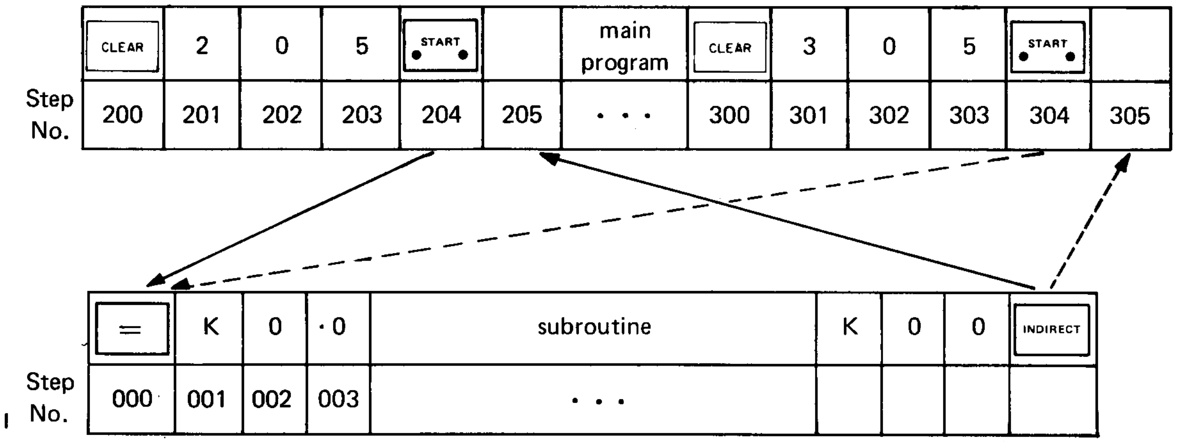
Note that) is used in step 096 to accumulate and display a result, but to prevent its storage.

2.3.2 Using Subroutines Stored in the Same Block as the Main Program

Subroutines requiring jumpering of decisions, or nesting of several subroutines require setting aside part of the program block for the subroutine (or subroutines). To jumper to the subroutine, use a go-to step to the beginning address of the subroutine. The return jumper from the subroutine back to the main program is accomplished using jumpering with a variable address, using the INDIRECT key.

When INDIRECT is encountered in a program, a jump takes place to the location designated by the last three digits in the calculator display. In the following example, the subroutine is located at step 000 and addressed with START as the program step, since START is equivalent to the sequence GO TO 000.

Observe in the following example that the displayed 205 (and 305) is stored in K_{00} at the beginning of the subroutine, and put back in the display at the end of it.



	CLEAR	2	0	5	START		main program	CLEAR	3	0	5	START	
Step No.	200	201	202	203	204	205	• • •	300	301	302	303	304	305

	=	K	0	·0	subroutine	K	0	0	INDIRECT
Step No.	000	001	002	003	• • •				

2.3.3 Setting a Flag for External Control

An example of the use of $\boxed{\text{IF FLAG}}$ is shown in the following keying sequence to find the average of n data points:

Press $\boxed{\text{END}}$ $\boxed{\text{LEARN}}$ $+ K_{10} = K_{10}\ K_{00} + 1 = K_{00}$ $\boxed{\text{CLEAR DISPLAY}}$

$\boxed{\text{IF FLAG}}$ 026 $\boxed{\text{GO TO}}$ 039 $K_{10} \div K_{00} = K_{05}$

Step 026

$\boxed{\text{REMOTE}}$ 9 $\boxed{\text{CLEAR}}$ $\boxed{\text{END}}$ $\boxed{\text{LEARN}}$

Step 039

Next Press $\boxed{\text{CLEAR DISPLAY}}$ $= K_{00} = K_{10}$

Enter x_1 $\boxed{\text{START}}$ x_2 $\boxed{\text{START}}$ . . . . x_n $\boxed{\text{SET FLAG}}$ $\boxed{\text{START}}$

The lights on $\boxed{\text{SET FLAG}}$ automatically go out.

2.4 USING f(x) AS A SUBROUTINE

All keys of the Scientist 909 and Statistician 911 are programmable, including f(x). When f(x) appears in a program it is executed. Thus, if a calculation is required in several places in a program, or if a calculation must be easily accessed for change, use f(x) as a subroutine.

Example: Integration of f(x) using Simpson's Rule, which is

$$\int_a^b f(x)\,dx \approx \frac{h}{3}\left(f(a) + 4\,f(a+h) + 2\,f(a+2h) + \dots + 2\,f(a+(m-2)h) + 4\,f(a+(m-1)h) + f(b)\right) = K_{10}$$

where the number of intervals $m = \frac{b-a}{h}$, an even natural number.

Programming Steps:

$\boxed{\text{END}}\ \boxed{\text{LEARN}}\ K_{02} - K_{01} = \div K_{03} = K_{00}$	(calculation of h), where a = K_{01}, b = K_{02}, m = K_{03}
$K_{01}\ \boxed{\text{f(x)}} = K_{10}\,K_{02}\ \boxed{\text{f(x)}} + K_{10} = K_{10}$	(calculation of f(a) + f(b))
$\boxed{\text{CLEAR}}\ 1 = K_{04}\ \boxed{\text{CLEAR}}\ 2 = K_{02}$	(initialization of counter and coefficient)
$\boxed{\text{CLEAR}}\ 6 - K_{02} = K_{02}\ K_{04} + 1 = K_{04}$	(calculation of 4, 2, 4 . . . . and counter)
$K_{01} + K_{00} = K_{01}\ f(x) \times K_{02} + K_{10} = K_{10}$	(summation)
$K_{04} - K_{03})\text{jump}\ \boxed{\text{IF DATA}-}\ 048\ K_{10} \times K_{00} \div 3 =$ $K_{10}\ \boxed{\text{END}}\ \boxed{\text{LEARN}}$	(summation)

Next, to integrate $e^{\sin x}$ from 0 to π with the number of intervals equal to 100, enter:

$\boxed{\text{DEFINE } \bullet f(x) \bullet}$ sin e^x $\boxed{\text{DEFINE } \bullet f(x) \bullet}$ $\boxed{\text{CLEAR DISPLAY}}$ $= K_{01}$ $\pi = K_{02}$ $\boxed{\text{CLEAR DISPLAY}}$ $100 = K_{03}$ $\boxed{\bullet \text{ START } \bullet}$,

and the value of the integral (6.208758036) is immediately displayed.

The Program is:

$\boxed{\bullet \text{ LEARN } \bullet}$ $= K_{05} + K_{01} = K_{01}\ K_{05}\ x^2 + K_{04} = K_{04}$ $\boxed{\text{CLEAR}}$ $\boxed{\bullet \text{ STOP } \bullet}$

$= K_{06} + K_{02} = K_{02}\ K_{05} \text{ X } K_{06} + K_{03} = K_{03}\ K_{00} + 1 = K_{00}$ $\boxed{\text{CLEAR}}$ $\boxed{\bullet \text{ STOP } \bullet}$

$K_{03} - K_{01} \text{ X } K_{02} \div K_{00} = \div\ (K_{04} - K_{01}\ x^2 \div K_{00}) = K_{11}$ $\boxed{\bullet \text{ STOP } \bullet}$

$K_{02} \div K_{00} - K_{11} \text{ X } K_{01} \div K_{00} = K_{12}$ $\boxed{\text{END}}$ $\boxed{\bullet \text{ LEARN } \bullet}$

$\boxed{\text{CLEAR}}$ $= K_{00} = K_{01} = K_{02} = K_{03} = K_{04}$

2.5 EXECUTING A PROGRAM

Once a program is stored in the Programmer internal memory, the user need only enter data and read answers. For example, to find the least squares line y = ax + b for a set of n data pairs, load the program and observe the following operating sequence:

Keying Sequence:	enter x_1 START ● ●	enter y_1 CONTINUE END	enter x_2 START ● ●	enter y_2 CONTINUE END	....	x_n START ● ●	y_n CONTINUE	CONTINUE	CONTINUE
Display Reads	0	0	0	0	....	0	0	a	b

The calculations to be made are:

$$a = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - x)^2} = \frac{\sum x_i y_i - \frac{1}{n} \sum x_i \sum y_i}{\sum x_i^2 - \frac{1}{n} \left(\sum x_i\right)^2}$$

$$= \frac{K_{03} - \frac{1}{K_{00} K_{01} K_{02}}}{K_{04} - \frac{1}{K_{00} (K_{01})^2}} = K_{11}$$

$$b = \bar{y} - a\bar{x} = K_{12}$$

where $\sum = \sum_{i=1}^{n}$, $\bar{x} = \frac{\sum x_i}{n}$, $\bar{y} = \frac{\sum y_i}{n}$

2.6 USING THE MAGNETIC TAPE CARTRIDGE

Tektronix program tape cartridges are loaded with high quality digital magnetic tape. To assure maximum life and performance from the cartridges, follow these recommendations:

- Avoid touching the exposed tape
- Do not leave the cartridge in the Programmer when not in use
- Store the cartridges in protective lint-free bags such as those in which they are shipped from Tektronix

The normal lifetime of each tape cartridge is in excess of 5000 revolutions. When a tape is no longer usable, it will not respond to a SEARCH PRGM or RECORD command. This lack of response occurs before data is lost or modified, thus assuring reliability of a "live" tape.

2.6.1 Recording a Program

Programs are recorded from the Programmer internal memory by the keying sequence RECORD followed by the one-digit block number of the block to receive the program. Recording erases all previously recorded information from the block. If pressed during the few seconds that the machine is recording, STOP is not learned and has no effect.

2.6.2 Searching the Magnetic Tape

The magnetic tape is organized in ten blocks of 512 steps per block. Blocks are identified 0 through 9. To find a program, press END to reset the counter to 000, then SEARCH PRGM followed by the one-digit address. This accesses the specified block on the cartridge and reads it into the internal memory. When the search process is started and the magnetic tape drive is coming up to speed, the Programmer search mechanism ignores several blocks to assure accurate reading of data. The number of ignored blocks varies slightly from machine to machine. To minimize search time, the following chart illustrates the order of fastest accessing.

Block previously accessed	0	1	2	3	4	5	6	7	8	9
Block next accessed	5	6	7	8	9	0	1	2	3	4

For example, search for block 5 (or 6) when starting from block 0. An efficiently organized program search will access and load a program block in a few seconds. Worst case, a program search can take 28 seconds.

To discontinue a program search, press LEARN to take the machine out of the learn mode, then STOP. Do not pull the cartridge out or otherwise move it while it is searching, since this movement can cause unwanted codes to be stored. The LEARN STOP sequence is required after SEARCH PRGM (not after RECORD). The sequence prevents STOP from being stored in the Programmer internal memory.

2.6.3 Storing Several Programs in One Block

More than one program can be recorded on a single tape block, with no upper limit as to the number of programs. Total program steps, however, cannot exceed 512 per block. Simply keep track of the step number of the beginning step in each program, and allow initial space in the program for indirect addressing.

2.6.4 Accessing One of Several Programs Within a Block

Program retrieval from a block containing more than one program is straight-forward. For example: four programs are stored in block 5 of a cartridge.

The step at which each program begins is:

Program	Step Number
1	025
2	078
3	400
4	450

The following programming steps provide an easy procedure for accessing the desired program.

X	4	+	4	9	2	)	INDIRECT	Main Program
000	001	002	003	004	005	006	007	

GO TO	0	2	5	GO TO	0	7	8	GO TO	4	0	0	GO TO	4	5	0
496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511

To start execution with program x, press x START. If the program block must first be read from the tape, press x SEARCH PRGM 5, and execution starts.

2.7 PRINTING

The address of the peripheral Tektronix Printer 941 is 9. To print the number appearing in the display, press REMOTE 9 on the calculator. The Programmer LIST key (preceded by a 9 in the calculator display) produces a program listing consisting of program step numbers, key codes (see APPENDIX A) and tape block number. To start a listing at a step other than 000, press GO TO, the starting address, then CLEAR 9 LIST. To stop a listing before it is finished, press STOP and hold it down for a few seconds to assure that the listing does not begin again after the key is released.

APPENDIX A

KEY CODES AND KEY ABBREVIATIONS

All keys of the Scientist 909 and Statistician 911 are programmable, and all Programmer 926 keys except $\boxed{\text{● LEARN ●}}$, $\boxed{\text{● RECORD ●}}$, $\boxed{\text{DISPLAY}}$, $\boxed{\text{STEP}}$, $\boxed{\text{LIST}}$ and $\boxed{\text{EDIT}}$ are programmable by the Programmer. Use of the Card Reader 923 permits all keys of all instruments to be programmed. 911 functions not accessible on the 909 keyboard are accessible, as are 909 functions not accessible on the 911. Additionally, logic is contained in both calculators for functions accessible only with the Card Reader. These are the decrementer K_{03} X (–.1) = K_{03},and the product accumulator Display X K_{04} = K_{04}.

The following abbreviations are used in the Program Library furnished with each Tektronix calculator.

Code	Key	Abbreviation	Code	Key	Abbreviation
000	0		026	(	
001	1		027	ln x	ln
002	2		030	REMOTE	R
003	3		031	$\sqrt{x}$	
004	4		032	x^2	
005	5		033	$\frac{1}{x}$	1/x
006	6		034	e^x	
007	7		035	log x	log
010	8		036	sin x	sin
011	9		040	DEFINE ● f(x) ●	Def
020	DEG ● ● RAD	D/R	041	–	
021	+		042	$\sqrt{x^2+y^2}$	$\sqrt{\bullet\bullet}$
022	$\lvert x\rvert^y$		043	÷	
023	×		044	arc	
024	=		045	+⟲–	+/–
025	.		046	cos x	cos

Code	Key	Abbreviation
047	CLEAR DISPLAY	CD
050[a]	tan x	tan
051[a]	π	
052	f(x)	
053	$K_{()}$	K
054[a]	hyper	hyp
055	int x	int
056	$x10^{oo}$	x10
057	CLEAR	C
060	)	
064[b]	ΣK_{00}	Σ_0
065[b]	ΣK_{01}	Σ_1
066[b]	$K_{02} + K_{03}$ $\cdot K_{02}$	Σ

Code	Key	Abbreviation
067[d]	K_{03} X (–.1) = K_{03}	Δ
070[d]	Display X K_{04} = K_{04}	π_4
071[b]	X !	
101[c]	IF DATA +	+
102[a]	IF DATA 0	0
103[c]	IF DATA –	–
104[c]	GO TO	G
105[c]	IF FLAG	F
106[c]	SET ●FLAG●	Sf
107[c]	IF RANGE	R
110[c]	PAUSE	P
111[c]	CONTINUE	Co
112[c]	INDIRECT	I
113[c]	SEARCH ●PRGM●	Sp
114[c]	END	E
115[c]	START ● ●	St
116[c]	RECORD ● ●	
132[c]	LIST	
133[c]	DISPLAY	
135[c]	LEARN ● ●	
136[c]	EDIT	
137[c]	STEP	
177[c]	STOP ● ●	S

a – Scientist 909 Keys

b – Statistician 911 Keys

c – Programmer 926 Keys

d – Functions Accessible Only by Card Reader 923

Note: Where no abbreviation is indicated, the key symbol itself is used.

APPENDIX B

INTRODUCTORY EXAMPLE PROGRAM

Interconnect and turn on the calculator and programmer, and try the following program to get the "feel" of the system.

GO TO 000 LEARN $K_{00} + 1 = K_{00}$ PAUSE $- K_{01} = K_{02}$ IF DATA – 000 K_{01} END LEARN

CLEAR $= K_{00}$ $10 = K_{01}$ START

Explanation of the Keystrokes:

GO TO 000 Commands the machine to go to the first step in the internal storage.

LEARN Tells the machine that the steps which follow are to be remembered.

$K_{00} + 1 = K_{00}$ Increments the number in K_{00} by 1.

PAUSE Causes the machine to stop long enough for the new number in K_{00} to be viewed.

$-K_{01} = K_{02}$ Subtracts the number in K_{01} from that in K_{00} and places the result in K_{02}.

IF DATA – 000 Commands the machine to jump to 000 if the number in K_{02} is negative.

K_{01} Recalls the number in K_{01} (and displays it), providing the machine did not jump at the previous step.

END	Terminates the program run.
LEARN	Tells the machine that no more steps are to be remembered.
CLEAR	Clears the display and any calculation that was in progress.
$= K_{00}$	Initializes K_{00} to 0.
$10 = K_{01}$	Initializes K_{01} to 10.
START	Begins program execution.

To record this program, select a cartridge block for recording, such as block 4. Press RECORD 4 and wait until the RECORD key lights go out. To retrieve the program, press END SEARCH PRGM 4 and wait until the SEARCH PRGM key lights go out.

APPENDIX C

PROGRAMMER 926 SPECIFICATIONS

Program Steps:	512
Operating Temperature:	0°C to 50°C
Storage Temperature:	−25°C to 85°C (tapes not included)
Power Requirements:	115 V/230 V (slide switch) ±10%, 50/60 Hz, 45 watts
Dimensions:	8 1/4 inches high X 8 1/4 inches wide X 15 1/4 inches deep
Weight:	16 pounds
Shipping Weight:	25 pounds
Magnetic Tape Cartridge 947 (Unprogrammed):	Endless loop, 10 block, 1/4 inch data grade magnetic tape, each block capable of storing up to 512 steps. (One 947 tape with diagnostic program supplied with Programmer.)