

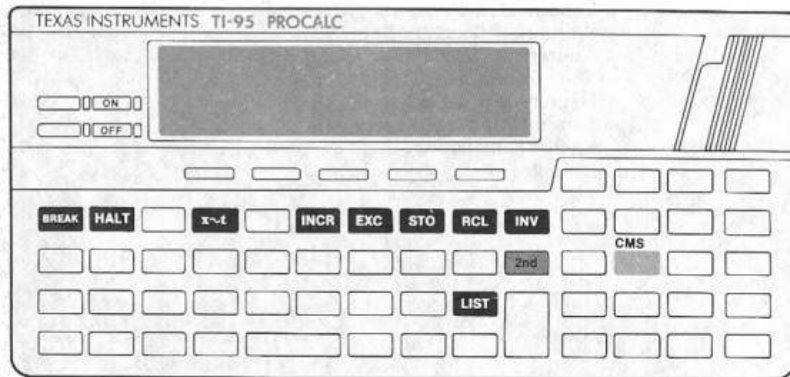
Chapter 5: Memory Operations

By using the data registers of the calculator, you can store, recall, and perform other operations on large amounts of numeric data. Data that you store in data registers is not lost when you turn the calculator off.

Table of Contents	Location of the Memory Keys	5-2
	Introduction	5-3
	Clearing Data Registers	5-4
	Storing and Recalling Data	5-5
	Addressing Methods	5-6
	The t-Register	5-7
	Memory Arithmetic	5-8
	Incrementing and Decrementing a Register	5-10
	Exchanging Values	5-11
	Listing	5-12

Location of the Memory Keys

The keys used to perform memory operations are shown in the figure below. Familiarize yourself with these keys and their location on the keyboard.



Introduction

You can use a segment of the TI-95 memory to hold numeric values while you perform calculations with other values. You can also perform some math operations on values stored in memory.

Types of Memory

The TI-95 memory is divided into data registers, program memory, and file space. Data registers are discussed in this chapter. Program memory and file space are discussed in the *TI-95 Programming Guide*. Although you can specify the amount of memory allocated to each of the three types of memory, the default allocations are adequate for most needs.

Data Registers

The first time you turn the calculator on, 125 data registers (numbered 000 through 124) are available for storage.

Each data register can hold a numeric value or alphanumeric characters. This chapter discusses only storing, recalling, and manipulating numeric data. Operations involving alphanumeric characters are discussed in the *TI-95 Programming Guide*.

Clearing Data Registers

Before beginning a task in which you will use data registers, you should consider clearing (setting to zero) the contents of the registers you will use. You can clear all data registers or a single data register.

Clearing All Registers

The key sequence $\boxed{2nd} \boxed{[CMS]}$ enables you to set to zero all memory currently allocated to data registers. Because this function clears the contents of **all** data registers, you should ensure there is no vital data stored in any registers before using it.

Clearing a Single Register

By storing a 0 in a data register, you can clear that register without affecting the contents of other registers.

Storing and Recalling Data

Storing and recalling numbers are two of the most often-used memory functions.

Storing Data

The $\boxed{[STO]}$ key followed by the data register address stores the currently displayed value in the register.

Because storing a value does not complete any pending operations, you can store a value you are entering into an expression or an intermediate result.

Example

Calculate 3×4.5^2 and store the intermediate result of 4.5^2 in data register 004.

Procedure	Press	Display
Clear display	$\boxed{[CLEAR]}$	0.
Begin expression	3 $\boxed{[x]}$	3.
Evaluate 4.5^2	4.5 $\boxed{[x^2]}$	20.25
Store 20.25 in register 004	$\boxed{[STO]} 004$	20.25
Display result	$\boxed{[=]}$	60.75

Recalling Data

The $\boxed{[RCL]}$ key displays the value stored in a specified register. (You can use the recall function to enter a stored value into an expression.)

Example

Multiply by 6 the value previously stored in data register 004.

Procedure	Press	Display
Clear display	$\boxed{[CLEAR]}$	0.
Begin expression	6 $\boxed{[x]}$	6.
Recall stored value	$\boxed{[RCL]} 004$	20.25
Display result	$\boxed{[=]}$	121.5

Addressing Methods

Because there are many data registers, each memory function requires you to specify which register you wish to use. You can use several methods to address a particular data register. If you make a mistake while entering the address or any other field, press **HELP**, **CLEAR**, or **OFF / ON** to cancel the entry, and then reenter the field correctly.

Long-Form Addressing

In the examples on the previous page, a three-character field is used to address a particular data register (004). When you use such long-form addressing, you must include all three characters, including leading zeros for addresses less than 100.

Short-Form Addressing

When referring to a data register whose address is less than 100, you can reduce keystrokes by omitting any leading zeros in the field. The calculator supplies the leading zeros when you press any key other than **HELP**, **CLEAR**, **OFF / ON**, or a number (digit) key.

For example, to use short-form addressing to recall the contents of data register 004 and add 12.3, press **RCL** 4 **+** 12.3 **=**. (When you press **+**, the 4 is accepted as the entire field for data register 004.)

Alphabetic Addressing

You may find it easier to remember an alphabetic letter than a number. You can use the letters A through Z to refer to the first 26 data registers (000 through 025). When you use such alphabetic addressing, the memory operation is completed as soon as you enter the letter.

For example, to recall the contents of data register D (register address 003), press **RCL** D.

Indirect Addressing

An addressing method used primarily in programming is indirect addressing. This method lets you refer to a particular data register by storing its register address in a second data register.

Indirect addressing is discussed in detail in the *TI-95 Programming Guide*.

The t-Register

Although the t-register (temporary register) is used primarily in statistics and polar/rectangular conversions, you can use it for temporary storage of a numeric value. When using the t-register for temporary storage, you should be aware of some characteristics that make this register different from a data register.

Using the t-Register

The **x \leftrightarrow t** key enables you to quickly exchange (swap) the displayed value with the value in the t-register. Because this operation requires only one keystroke to store or recall a value, you may occasionally find it more convenient than using a data register.

Example

Using the t-register, calculate $2.77^{3.55-2.77} + 3.55$.

Procedure	Press	Display
Clear display	CLEAR	0.
Begin calculation	2.77 y^x	2.77 y^x
Store 2.77 in t-register and continue problem	x\leftrightarrowt () 3.55 -	3.55 -
Store 3.55, restore 2.77 to display	x\leftrightarrowt	2.77
Perform subtraction, then exponentiation) +	2.213778778 +
Restore 3.55 to display	x\leftrightarrowt	3.55
Display result	=	5.763778778

Limitations

The operation of the t-register is different from that of a data register. Remember the following limitations.

- ▶ You lose any value stored in the t-register if you turn off the calculator or perform any functions that use the t-register, such as statistics functions or polar/rectangular conversions.
- ▶ You cannot perform memory arithmetic, increment, or decrement operations using the t-register.

Memory arithmetic can reduce keystrokes by letting you both perform a calculation and store its result using a single operation. Because memory arithmetic does not disturb the original displayed value and does not complete any pending operations, you can use it while entering an expression without affecting the way the expression is evaluated.

Data Register Addition The key sequence **[STO] [+]** followed by the address of a data register adds the displayed value to the contents of the specified register.

For example, if the display contains the value 1000 and data register 000 contains the value 234, the key sequence **[STO] [+]** 000 leaves the sum 1234 in data register 000.

Data Register Subtraction The key sequence **[STO] [-]** followed by the address of a data register subtracts the displayed value from the contents of the specified register.

For example, if the display contains the value 1000 and data register 000 contains the value 1234, the key sequence **[STO] [-]** 000 leaves the difference 234 in data register 000.

Data Register Multiplication The key sequence **[STO] [x]** followed by the address of a data register multiplies the contents of the specified register by the displayed value and stores the product in the register.

For example, if the display contains the value 1000 and data register 000 contains the value 234, the key sequence **[STO] [x]** 000 leaves the product 234000 in data register 000.

Data Register Division The key sequence **[STO] [+]** followed by the address of a data register divides the contents of the specified register by the displayed value and stores the quotient in the register.

For example, if the display contains the value 1000 and data register 000 contains the value 234000, the key sequence **[STO] [+]** 000 leaves the quotient 234 in data register 000.

Example

The following example demonstrates each of the four memory arithmetic functions.

Procedure	Press	Display
Clear display	[CLEAR]	0.
Store 25 in register 000	25 [STO] 000	25.
Add 215 to register 000 (register 000 now contains 240)	215 [STO] [+] 000	215.
Subtract 200 from register 000 (register 000 now contains 40)	200 [STO] [-] 000	200.
Multiply register 000 by 2 (register 000 now contains 80)	2 [STO] [x] 000	2.
Divide register 000 by 4 (register 000 now contains 20)	4 [STO] [+] 000	4.
Verify contents of register 000	[RCL] 000	20.

Incrementing and Decrementing a Register

You can add one to or subtract one from the contents of a register using the INCR function.

Increment and Decrement

The TI-95 includes specialized forms of data-register addition and subtraction. The **INCR** key increments the contents of a register by one and the key sequence **INV INCR** decrements the contents of a register by one.

Example

The following example demonstrates the use of the INCR function.

Procedure	Press	Display
Clear display	CLEAR	0.
Store 1234 in register 000	1234 STO 000	1234.
Increment register 000 by two using short-form addressing	INCR 0 INCR 0	INC _ 0 INC _ 0
Recall register 000 contents	RCL 000	1236.
Decrement register 000 by one using short-form addressing	INV INCR 0	INV INC _ 0
Recall register 000 contents	RCL 000	1235.

Exchanging Values

In addition to storing, recalling, and performing memory arithmetic, you can exchange the displayed value with a stored value.

Exchanging

The **EXC** key followed by the address of a data register exchanges the displayed value with the value stored in the specified register.

Exchanging values can eliminate the need to perform separate store and recall operations.

Example

Evaluate $x^2 + 4xy + 2y^2$ where $x = .25$ and $y = 1.2$. (This example uses alphabetic addressing.)

Procedure	Press	Display
Clear display	CLEAR	0.
Store x	.25 STO A	0.25
Evaluate x^2	x² +	0.0625 +
Enter y	1.2 x	1.2 *
Store y, recall x	EXC A	0.25
Evaluate $x^2 + 4xy$	x 4 +	1.2625 +
Recall y	2 x RCL A	1.2
Evaluate y^2	x²	1.44
Display result	=	4.1425

You may want to know the contents of a series of data registers or the status of the calculator. The LIST function provides you with this capability. You can also list programs and labels as explained in *The TI-95 Programming Guide*. If you have a PC-324 printer, you can print the listing. For more information, refer to Chapter 6, "Using an Optional Printer."

Listing Data Registers

To list the contents of a series of data registers:

1. Press **[LIST]**. The display shows the list options.

```
LIST:
REG PGM LBL ST
```

2. Enter the address of the first register to list. If you do not enter an address, the list begins at register 000 each time you press the **[LIST]** key.
3. Select <REG>.

Beginning with the specified register, the calculator lists the addresses and contents at a one-second rate.

If you do not have a printer connected, you can use the **[→]** key to control the speed of the displayed listing.

- ▶ To pause the listing indefinitely, hold down the **[→]** key.
- ▶ To advance through the listing without the one-second pause, repeatedly press and release the **[→]** key.

Listing the Calculator Status

Press **[LIST]** <ST> to list the current calculator status.

The calculator lists the current partition settings and each parameter that is not currently set to its default setting. (Parameters that are currently set to their default settings are not listed.)

The list is displayed at a one-second rate. You can use the **[→]** key to control the speed of the listing.

If you have a PC-324 printer attached, the listing is also printed. For more information, refer to Chapter 6, "Using an Optional Printer."

The list below shows all possible statuses.

Partition (memory)
Angle mode (degrees, radians, or grads)
Hexadecimal mode (with or without two's complement)
Octal mode (with or without two's complement)
Flags set (flag numbers)
Fixed decimal (places)
Module (installed)
Word break mode (if turned on)
Print device (number)
Print width (number)
System mode (if selected)
Last error (since **[LIST]** <ST> function last used)

Stopping the Listing

To stop a listing before all register contents or status messages have been displayed, hold down the **[BREAK]** or **[HALT]** key until the word **LIST:** is displayed.

After you stop a listing, you can perform other operations or you can list the registers again by selecting <REG>. The list begins at the number in the display, register. If an alpha message is in the display, you can press **[CE]** to see the number.