



hp calculators

HP 33S Using Register Arithmetic

Variables and Memory Registers

Practice Examples:

Calculating Using STO Arithmetic

Calculating Using RCL Arithmetic

Storage Arithmetic in a Program



Variables and Memory Registers

The purpose of this training aid is to show how arithmetic can be carried out directly using HP 33S memory registers. First, the following is a short explanation of variables and registers; the same information is given in the practice aid on using calculator memories. That training aid then gives more information on the various other uses of memory registers.

When an equation is typed on the HP 33S, it can use variables with names from A through Z. For example an equation

$$3X^2 - 5X = A$$

has the variables X and A in it.

Variables can also be used in programs and in calculations from the keyboard.

Each variable consists of a number and of a place in the calculator memory where the number is stored.

The number is called the value of the variable. If no value has been given to a variable then its value is 0.

The place in memory where this number is stored is called a data register, or a memory register, or just a register.

Each memory register can be referred to by a numeral as well as by its name. Register A is 1 and register Z is 26. Seven more registers can be referred to by numerals, six hold values from statistics calculations and one is a special "index" register. The names "data register" or "memory register" or "memory" refer to all of these, not just the variables, so these names are sometimes used in this training aid, rather than the name "variables."

The training aid on general uses of registers shows that numbers can be stored in registers by the STO command, and can be recalled from registers by the RCL command. This aid covers the special topic of how arithmetic can be carried out directly with memory registers using the commands STO+, STO-, STO×, STO÷, RCL+, RCL-, RCL× and RCL÷.

Practice Example: Calculating Using STO Arithmetic

The HP 33S provides a set of physical constants, such as the speed of light. Users who need to store other constants can put them into the memory registers so they can be used in calculations.

Example 1: A planetary scientist needs to perform several calculations using the masses of the Earth and the Moon. First the scientist wants to store the mass of the Earth in register E and of the Moon in M, in kilograms. The Earth-Moon system is sometimes called a "twin planet system" and the scientist wants to put the total mass of the Earth and the Moon in register T.

Solution: First type the Moon's mass, then press **[STO]** and the variable name to store the mass in that variable's memory register. Press the keys as shown below. The key labeled **[E]** is above the **[9]** on the keyboard. Typing **[E][2][2]** means "times 10 to the power 22".

[7][.][3][4][9][E][2][2][STO][M]

When **[STO]** is pressed, the symbol "A..Z" appears at the top of the screen. This tells the user that the next key pressed should be one of the keys with letters A to Z at their lower right, and that the corresponding

letter will be used. For the letter “M”, press the $\boxed{\text{TAN}}$ key. The number is stored in M, but remains on the lower line of the screen, as shown in Figure 1.



Figure 1

Store the Moon’s mass in register T as well, so that the Earth’s mass can be added to it. The mass has already been stored in M but it is still available to be stored in T as well.

$\boxed{\text{STO}}$ $\boxed{\text{T}}$

Now type the mass of the Earth and store it in register E. To use register E, press $\boxed{\Sigma+}$ when the A..Z symbol is at the top of the screen. This is not the same as the key labeled $\boxed{\text{E}}$ used for entering powers of 10.

$\boxed{5}$ $\boxed{\cdot}$ $\boxed{9}$ $\boxed{7}$ $\boxed{3}$ $\boxed{6}$ $\boxed{\text{E}}$ $\boxed{2}$ $\boxed{4}$ $\boxed{\text{STO}}$ $\boxed{\text{E}}$

The mass of the Earth is now in register E and is still on the lower line of the screen. To add it to the mass of the Moon already in register T, use the STO+ command

$\boxed{\text{STO}}$ $\boxed{+}$ $\boxed{\text{T}}$

This takes the number in the lower line of the display and adds it to register T. The number is still available for further use. It is still the “current number”, or in RPN mode, the number is still in stack register X.



Figure 2

Answer: The mass of the Moon is now in register M, the mass of the Earth is in register E, and the mass of the Earth-Moon twin system is in register T. The mass of the Earth is also still in the current register, as shown in Figure 2. To confirm that the number in register T is the sum of the two masses, view register T by pressing:

$\boxed{\text{R}}$ $\boxed{\text{VIEW}}$ $\boxed{\text{T}}$



Figure 3

The STO+ command adds the current number to the register selected by its name. In the same way, STO- subtracts the current number from the named register, STO \times multiplies the named register by the current number and STO \div divides the named register by the current number. In all cases, the number itself is unchanged and continues to be the current number. As it has not changed, the value in the LastX register also remains unchanged.

Practice Example: Calculating Using RCL Arithmetic

Example 2: If the Earth-Moon system is called a twin system, the mass of the Moon should be fairly similar to the mass of the Earth. How do they compare?

Solution: The mass of the Earth is still in the current register. Recall the mass of the Moon from register M and divide it into the current register. Press these keys.

RCL \div **M**

The result of the division is now in the current register.

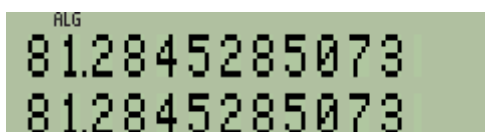


Figure 3

Answer: The ratio of the masses of the Earth and the Moon is about 1:81. The Moon's mass is only about 1.2% that of the Earth's, not really a twin. Nevertheless, the two are much closer in mass than any other major planet-satellite system. For example the mass of Ganymede is 0.0078% of the mass of Jupiter, and the mass of Deimos is 0.0000017% of the mass of Mars. Only Charon and Pluto are closer, with Charon having about 15% of the mass of Pluto – and not all planetary scientists are willing to consider Pluto to be a planet.

When **RCL** is pressed, followed by an arithmetic operation and a letter, then the number in the chosen register is recalled and added to, subtracted from, multiplied by, or divided into, the current register. As opposed to STO arithmetic, RCL arithmetic changes the current number and leaves the stored number unchanged. Because the current number is changed, its previous value is stored in the LastX register for re-use in RPN mode.


Note: In algebraic mode, the upper line of the screen shows the calculation performed, and the lower line shows the result. When a number is recalled from a register, the register name is shown on the upper line and the recalled value is shown on the lower line. When a RCL calculation is carried out in algebraic mode, its result is displayed on the upper line, and is shown on the lower line too, as it is the new current number. This can be seen in Figure 3.

Practice Example: Storage Arithmetic in a Program

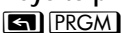










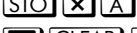

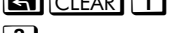

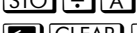


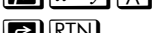



As these examples show, STO and RCL arithmetic make some calculations easier to carry out. It is generally quicker to use a STO or RCL arithmetic operation rather than to RCL a number, calculate with it, and then STO the result. Quite apart from this, many users find that storage register arithmetic has important applications in programs. There are three reasons for this.



- (1) Each STO or RCL instruction takes one less step in a program than STO or RCL followed by a separate arithmetic command. Saving a step in a program makes it shorter, faster and easier to read.
- (2) STO and RCL arithmetic works in the same way in RPN mode and in algebraic mode. This means that programs can be written that will work both in RPN and in algebraic mode. Such programs are very helpful if an HP 33S owner likes to use both modes, or if a program is written for other people who might use the calculator in either mode.







- (3) STO arithmetic does not use the RPN stack and does not change the LastX register. This means that programs can be written to work in the same way as built-in HP 33S functions, to replace the original number with a calculated result, keep a copy of the original number in LastX, and leave the rest of the stack unchanged. Such programs can then be used like built-in functions, and can be called from other programs.

Example 3: The planetary scientist wants a program to calculate the volume of a planet, given its radius R. The program should work both in RPN mode and in algebraic mode, it should not change the rest of the stack in RPN mode, and it should leave the LastX register unchanged, so R can be recalled by  LASTX.

Solution: Assuming that a planet is spherical, the volume is given by $\frac{4}{3} \pi R^3$. The program below calculates this formula using register arithmetic. The use of CLx is to remove values without changing the stack. Press these keys.

Keys to press	Prgm line	Command	Explanation
			select program entry mode
			go to top of program memory
 V	V0001	LBL V	put a label at the beginning, V for Volume
	V0002	x³	cube R and also save it in LastX
 A	V0003	STO A	put R³ in register A
 1	V0004	CLx	remove R³
	V0005	π	
 X 	V0006	STO× A	multiply R³ in A by π
 1	V0007	CLx	remove π
	V0008	4	
 X 	V0009	STO× A	multiply A by 4
 1	V0010	CLx	remove 4
	V0011	3	
 ÷ 	V0012	STO÷ A	divide A by 3
 1	V0013	CLx	remove 3 so zero can be put in A
 	V0014	x↔ A	Put 0 in A and A's current value is $\frac{4}{3} \pi R^3$ in X
	V0015	RTN	end the program
			leave program entry mode

Programming is explained in more detail in the training aid on writing a simple program. That training aid explains how a program's length and checksum can be found. The program above has a length of 69 and checksum 4CD2. To try the program, type the equatorial radius of the Moon in km and press  .

The result is shown in the current register.

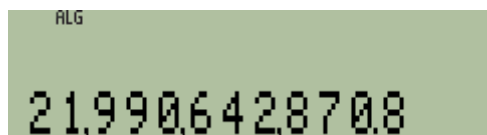


Figure 4

This is a very large number, so press **ENG** to see the result with an appropriate power of 10. The **ENG** key is the large silver key at the top left directly below the calculator screen.



Figure 5

The same result is obtained in algebraic and in RPN modes. In RPN mode, stack registers Y, Z and T are unchanged, and LastX brings back the radius. Other programs can use **XEQ** **V** to execute this program like a function, whenever the volume of a sphere needs to be calculated.

Answer: The volume of the Moon is calculated as 2.199E10 cubic kilometers. The true value is close to 2.196E10 because the Moon is not perfectly spherical. The error is about 0.1%. It would be far bigger for Saturn, which is much more flattened at the poles.

More complicated programs can use the index register to carry out STO and RCL arithmetic on several registers, with commands such as STO+ (i). Use of the index register is described in the training aid on general uses of memory registers.

This training aid has shown how STO and RCL arithmetic can be useful in keyboard calculations and in programming. With experience, users can find many occasions where storage arithmetic on the HP 33S is a real help in their work.