

group	func	desc
alpha	AIP	<i>Append Integer part</i> of x to the Alpha register.
alpha	ALENG	<i>Alpha length</i> . Returns the number of characters in the Alpha register.
alpha	AOFF	<i>Alpha off</i> . Exit from the ALPHA menu.
alpha	AON	<i>Alpha on</i> . Select the ALPHA menu.
alpha	ARCL	<i>Alpha recall</i> . Copy the first six characters in the Alpha register from a register or variable. Parameter: register or variable (indirect allowed)
alpha	AROT	<i>Alpha rotate</i> . Rotate the Alpha register by the number of characters specified in the x-register.
alpha	ASHF	<i>Alpha shift</i> . Shifts the six left-most characters out of the Alpha register.
alpha	ASTO	<i>Alpha store</i> . Copy the first six characters in the Alpha register into a register or variable. Parameter: register or variable (indirect allowed)
alpha	ATOX	<i>Alpha to X</i> . Convert the left-most character in the Alpha register to its character code (returned to the x-register) and delete the character.
alpha	AVIEW	<i>Alpha view</i> . Display the Alpha register.
alpha	CLA	<i>Clear Alpha register</i> . If Alpha mode is on and character entry is terminated (no cursor displayed), then ◀ also executes the CLA function.
alpha	POSA	<i>Position in Alpha</i> . Searches the Alpha register for the target specified in the x-register. If found, returns the character position; if not found, returns -1.
alpha	PRA	Print Alpha register.
alpha	PROMPT	Display the Alpha register and halt program execution.
alpha	STR?	If the x-register contains an Alpha string, execute the next program line; if the x-register does not contain an Alpha string, skip the next program line.
alpha	XTOA	<i>X to Alpha</i> . Appends a character (specified by the code in the x-register) to the Alpha register. If the x-register contains an Alpha string, appends the entire string.
flag	CF	Clear flag nn (00 ≤ nn ≤ 35 or 81 ≤ nn ≤ 99). Parameter: flag number (indirect allowed)
flag	FC?	Flag clear test. If the specified flag is clear, executes the next program line; if the flag is set, skips the next program line. Parameter: flag number (indirect allowed)
flag	FC?C	Flag clear test and clear. If the specified flag is clear, execute the next program line; if the flag is set, skip the next program line. Cleared after the test is complete. (This function can be used only with flags 00 through 35 and 81 through 99.) Parameter: flag number (indirect allowed)
flag	FS?	Flag set test. If the specified flag is set, execute the next program line; if the flag is clear, skip the next program line. Parameter: flag number (indirect allowed)
flag	FS?C	Flag set test and clear. If the specified flag is set, execute the next program line; if the flag is clear, skip the next program line. Clear the flag after the test is complete. (This function can be used only with flags 00 through 35 and 81 through 99.) Parameter: flag number (indirect allowed)
flag	SF	Set flag nn (00 ≤ nn ≤ 35; 81 ≤ nn ≤ 99). Parameter: flag number (indirect allowed)
graf	AGRAPH	<i>Alpha graphics</i> . Display a graphics image. Each character in the Alpha register specifies an 8-dot column pattern. The x- and y-registers specify the pixel location of the image.
graf	PIXEL	Turn on a single pixel (dot) in the display. The location of the pixel is given by the numbers in the x- and y-registers.
I/O	BEEP	Sound a sequence of four tones.
I/O	EXITALL	<i>Exit all</i> menus.
I/O	GETKEY	<i>Get key</i> . The calculator waits for you to press a key. When you do, the key number is returned to the x-register. Keys are numbered from 1 through 37 (Σ+ through ÷) for normal keys and 38 through 74 (◀ through CATALOG) for shifted keys.
I/O	INPUT	Recall a register or variable to the x-register, display the name of the register or variable along with the contents of the x-register, and halt program execution. Pressing R/S (or SST) stores x into the register or variable; pressing EXIT cancels. (Used only in programs.) Parameter: register or variable (indirect allowed)
logic	AND	<i>Logical AND</i> . Returns x AND y.
logic	BIT?	Test the x ⁿ bit of y. If the bit is set (1), execute the next program line; if the bit is clear (0), skip the next program line.
logic	NOT	<i>Logical NOT</i> . Returns NOT(x).
logic	OR	<i>Logical OR</i> . Returns x OR y.
logic	ROTX	<i>Rotate</i> the 36-bit number in the y-register by x bits.
logic	XOR	<i>Logical XOR</i> (exclusive OR). Returns x XOR y.
print	ADV	<i>Advance</i> the printer paper one line
print	DELAY	Set the <i>print delay</i> time to x seconds.
print	LIST	Print a portion of a program listing. (Not programmable.) Parameter: number of lines.
print	MAN	Select <i>Manual print mode</i> .
print	NORM	Select <i>Normal print mode</i> , which prints a record of keystrokes.
print	PRLCD	<i>Print LCD</i> (liquid crystal display). Prints the entire display.
print	PROFF	<i>Printing off</i> . Clears flags 21 and 55.
print	PRON	<i>Printing on</i> . Sets flags 21 and 55.
print	PRP	<i>Print program</i> . If a label is not specified, print the current program. (Not programmable.) Parameter: global label (optional)
print	PRSTK	<i>Print stack</i> . Print the contents of the stack registers (x, y, z and t).
print	PRUSR	<i>Print user</i> variables and programs.
print	PRV	<i>Print variable</i> . Parameter: variable name (indirect allowed)
print	PRX	<i>Print x-register</i> .
print	PRZ	<i>Print statistics</i> . Prints the contents of the summation registers.
print	TRACE	Select <i>Trace</i> printing mode, which prints a record of keystrokes and results.
prog	BST	<i>Back step</i> . Move the program pointer to the previous program line. (Not programmable.)
prog	DEL	<i>Delete</i> the specified number of lines from the current program. Program-entry mode must be on. (Not programmable.) Parameter: number of lines.
prog	END	<i>End</i> of a program.
prog	ENTER	Separate two numbers keyed in sequentially; copies x into the y-register, y into the z-register, and z into the t-register, and loses t.
prog	GTO	<i>Go to label</i> . From the keyboard, move the program pointer to the specified label. In a running program, cause the program to branch to the specified label. Parameter: local or global label (indirect allowed)

prog	LBL	<i>Label</i> . Identify programs and routines for execution and branching. Parameter: local or global label.
prog	LCLBL	<i>Select Local label mode</i> for the CUSTOM menu (to use CUSTOM menu assignments to execute local labels within the current program).
prog	OFF	Turn the calculator off (programmable). (Pressing OFF does not execute the programmable OFF function.)
prog	ON	<i>Continuous on</i> . Prevent the calculator from automatically turning off after ten minutes of inactivity.
prog	PSE	<i>Pause</i> program execution for about 1 second.
prog	R/S	<i>Run/stop</i> . Runs a program (beginning at the current program line) or stops a running program. In program-entry mode, inserts a STOP instruction into the program.
prog	RTN	<i>Return</i> . In a running program, branches the program pointer back to the line following the most recent XEQ instruction. If there is no matching XEQ instruction, program execution halts. From the keyboard, RTN moves the program pointer to line 00 of the current program.
prog	STOP	<i>Stop</i> program execution. (R/S in program entry mode).
prog	TOONE	Sounds a <i>tone</i> . Parameter: tone number (0–9) (indirect allowed)
prog	XEQ	<i>Execute</i> a function or program. Parameter: function or label (indirect allowed)
prog loop	DSE	<i>Decrement, Skip if (less than or) Equal</i> . Given cccccc.fffii in a variable or register, decrements cccccc by ii and skips the next program line if cccccc is now \leq fff. Parameter: register or variable (indirect allowed)
prog loop	ISG	<i>Increment, Skip if Greater</i> . Given cccccc.fffii in a variable or register, increments cccccc by ii and skips the next program line if cccccc is now $>$ fff. Parameter: register or variable (indirect allowed)
prog menu	ASSIGN	<i>Assign</i> a function, program, or variable to a menu key in the CUSTOM menu. Parameter 1: function name, alpha program label, or variable name. Parameter 2: key number (01–18).
prog menu	KEYASN	Selects <i>key-assignments</i> mode for the CUSTOM menu.
prog menu	KEYG	<i>On menu key, go to</i> . Branch to specified label when a particular menu key is pressed. Parameter 1:: Key number (1 through 9), Parameter 2: program label (global or local)
prog menu	KEYX	<i>On menu key, execute</i> . Execute (as a subroutine) specified label when a particular menu key is pressed. Parameter 1:: Key number (1 through 9), Parameter 2: program label (global or local)
prog menu	MENU	Select the programmable menu.
prog menu	VARMENU	Create a <i>variable menu</i> using MVAR instructions following the specified global label. Parameter: global program label. (indirect allowed)
prog test	CPX?	If the x-register contains a complex number, execute the next program line; if the x-register does not contain a complex number, skip the next program line.
prog test	MAT?	If the x-register contains a matrix, execute the next program line; if the X-register does not contain a matrix, skip the next program line.
prog test	REAL?	If the x-register contains a real number, execute the next program line; if the x-register does not contain a real number, skip the next program line.
prog test	X<0?	<i>X less than zero test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X<Y?	<i>X less than y test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X=0?	<i>X equal to zero test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X#0?	<i>X not equal to zero test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X=Y?	<i>X equal to y test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X#Y?	<i>X not equal to y test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X>0?	<i>X greater than zero test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X>Y?	<i>X greater than y test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X≤0?	<i>X less than or equal to zero test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X≤Y?	<i>X less than or equal to y test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X≥0?	<i>X greater than or equal to zero test</i> . If true, execute the next program line; if false, skip the next program line
prog test	X≥Y?	<i>X greater than or equal to y test</i> . If true, execute the next program line; if false, skip the next program line
statistic	ALLΣ	Select ALLΣ (All-statistics) mode, which uses 13 summation coefficients.
statistic	BEST	Select the <i>best</i> curve-fitting model for the current statistical data.
statistic	COMB	<i>Combinations</i> of y items taken x at a time = $y! / [x!(y-x)!]$
statistic	CORR	Returns a correlation coefficient using the current statistical data and curve-fitting model.
statistic	EXPF	Select the <i>exponential</i> curve-fitting model.
statistic	FCSTX	<i>Forecasts an x-value</i> given a y-value.
statistic	FCSTY	<i>Forecasts a y-value</i> given an x-value.
statistic	LINΣ	Select <i>Linear statistics</i> mode, which uses six summation coefficients.
statistic	LINF	Select the <i>linear</i> curve-fitting model.
statistic	LOGF	Select the <i>logarithmic</i> curve-fitting model.
statistic	MEAN	<i>Mean</i> . Returns the mean of x-values ($\Sigma x / n$) and the mean of y-values ($\Sigma y / n$).
statistic	PERM	<i>Permutations</i> of y items taken x at a time. Returns $y! / (y - x)!$
statistic	PWRF	Select the <i>power</i> curve-fitting model.
statistic	RAN	Returns a <i>random</i> number ($0 \leq x < 1$)
statistic	SDEV	<i>Standard deviation</i> . Returns s_x and s_y using the current statistical data.
statistic	SEED	Store a <i>seed</i> for the random number generator.
statistic	SLOPE	Return the <i>slope</i> of the linear transformation of the current curve-fitting model.
statistic	SUM	Returns the sums Σx and Σy into the x- and y-registers.
statistic	WMEAN	<i>Weighted mean</i> . Return the mean of x-values weighted by the y-values $\Sigma xy / \Sigma y$
statistic	YINT	<i>y-intercept</i> . Returns the y-intercept of the curve fitted to the current statistical data.

statistic	$\Sigma-$	<i>Summation minus.</i> Subtract a pair of x- and y-values from the summation registers.
statistic	$\Sigma+$	<i>Summation plus.</i> Accumulate a pair of x- and y values into the summation registers.
statistic	Σ REG	<i>Summation registers.</i> Defines which storage register begins the block of summation registers. Parameter: register number (indirect allowed)
statistic	Σ REG?	Return the register number of the first summation register.
status	ALL	Select the <i>All</i> display format.
status	CLALL	<i>Clear all.</i> Clear all stored programs and data.(Not Programmable.)
status	CLD	<i>Clear display.</i> Clear a message from the display.
status	CLKEYS	Clear all CUSTOM menu key assignments.
status	CLLCD	<i>Clear LCD</i> (liquid crystal display). Blanks the entire display.
status	CLMENU	<i>Clear MENU.</i> Deletes all menu key definitions for the programmable menu.
status	CLP	<i>Clear a program</i> from memory. Parameter: global label
status	CLRG	<i>Clear Registers.</i> Clear all of the numbered storage registers to zero.
status	CLST	<i>Clear Stack.</i> Clear the stack registers to zero.
status	CLV	<i>Clear a variable</i> from memory. Parameter: variable name (indirect allowed)
status	CLX	<i>Clear x-register</i> to zero. If digit entry is terminated (no cursor in the display), then ◀ also executes CLX.
status	CLZ	<i>Clear statistics.</i> Clear the accumulated statistical data in the summation registers.
status	DECM	Selects <i>Decimal</i> mode (base 10).
	DEG	Select the <i>Degrees</i> angular mode.
status	ENG	Select <i>Engineering</i> display format. Parameter: number of digits (indirect allowed)
status	FIX	Select <i>Fixed-decimal</i> display format. Parameter: number of digits (indirect allowed)
status	GRAD	Select <i>Grads</i> angular mode.
status	HEXM	Select <i>Hexadecimal</i> mode (base 16).
status	OCTM	Select <i>Octal</i> mode
status	QUIET	Toggle flag 26 to disable or enable the beeper (Not programmable)
status	RDX,	Select a comma to be used as the radix mark (decimal point).
status	RDX.	Select a period to be used as the radix mark (decimal point).
status	SCI	Select <i>scientific</i> notation display format. Parameter: number of digits (indirect allowed)
status	SIZE	Set the number of storage registers. Parameter: number of registers.
util	→OCT	<i>To octal.</i> Converts a decimal number to the octal representation. Note: This function is included to provide program compatibility with the HP-41 (which uses the function name OCT).
util	→DEC	<i>To decimal.</i> Converts the octal (base 8) representation of a number to decimal (base 10). Note: This function is included to provide program compatibility with the HP-41 (which uses the function name DEC).
util	→DEG	<i>To degrees.</i> Convert an angle-value from radians to degrees. Returns $x \times (180/\pi)$.
util	→HMS	<i>To hours, minutes, and seconds.</i> Convert x from a decimal fraction to a minutes-seconds format.
util	→HR	<i>To hours.</i> Converts x from a minutes-seconds format to a decimal fraction.
util	→POL	<i>To polar.</i> Converts x and y to the corresponding polar coordinates r and θ . If the x-register contains a complex number, converts the two parts of the number to polar values.
util	→RAD	<i>To radians.</i> Converts a angle value in degrees to radians. Returns $x \times (\pi/180)$.
util	→REC	<i>To rectangular.</i> Converts r (in the x-register) and θ (in the y-register) to the corresponding rectangular coordinates, x and y. If the X-register contains a complex number, converts the two parts of the number to rectangular values.
util	◀	<i>Backspace</i> or clear x-register. In Program entry mode, deletes the current program line.
util	HMS-	Subtract x from y using H.MMSSss format.
util	HMS+	Add x and y using H.MMSSss (hours-minutes-seconds) format.
util	LASTX	<i>Last x.</i> Recall the last value of x used in a calculation.
util	R↑	<i>Roll up</i> the contents of the four stack registers one position.
util	R↓	<i>Roll down</i> the contents of the four stack registers one position.
util	VIEW	<i>View</i> the contents of a register or variable. Parameter: register or variable (indirect allowed)
util	X<>	Swaps the contents of the x-register with another register or variable. Parameter: register or variable (indirect allowed)
util	X<>Y	Swaps the contents of the x- and y-registers.
var	LSTO	<i>Store Local Variable.</i> Global var gets covered, local var gets deleted after RTN
var	RCL	<i>Recall</i> data into the x-register. Parameter: register or variable (indirect allowed)
var	RCL-	<i>Recall subtraction.</i> Recall data and subtract it from the contents of the x-register. Parameter: register or variable (indirect allowed)
var	RCL+	<i>Recall addition.</i> Recall data and add it to the contents of the x-register. Parameter: register or variable (indirect allowed)
var	RCL÷	<i>Recall division.</i> Recall data and divide it into the contents of the x-register. Parameter: register or variable (indirect allowed)
var	RCL×	<i>Recall multiplication.</i> Recall data and multiply it by the contents of the x-register. Parameter: register or variable (indirect allowed)
var	STO	<i>Store</i> a copy of x into a destination register or variable. Parameter: register or variable (indirect allowed)
var	STO÷	<i>Store division.</i> Divides an existing register or variable by x. Parameter: register or variable (indirect allowed)
var	STO×	<i>Store multiplication.</i> Multiplies an existing register or variable by x. Parameter: register or variable (indirect allowed)
var	STO-	<i>Store subtraction.</i> Subtracts x from an existing register or variable. Parameter: register or variable (indirect allowed)
var	STO+	<i>Store addition.</i> Adds x to an existing register or variable. Parameter: register or variable (indirect allowed)
math	-	<i>Subtraction.</i> Returns $y - x$.
math	%	<i>Percent.</i> Returns $(x \times y) / 100$. (Leaves the y value in the y-register.)
math	%CH	<i>Percent change.</i> Returns $(x - y) \times (100 / y)$.
math	+	<i>Addition.</i> Returns $y + x$.
math	+/-	<i>Change the sign</i> of the number in the x-register. While entering an exponent, can also be used to change the sign of the exponent.

math	÷	<i>Division.</i> Returns y / x .
math	×	<i>Multiplication.</i> Returns $x \times y$.
math	1/x	<i>Reciprocal.</i> Returns $1/x$.
math	10↑X	<i>Common exponential.</i> Returns 10^x .
math	ABS	<i>Absolute value.</i> Returns $ x $.
math	ACOS	<i>Arc cosine.</i> Returns $\cos^{-1} x$.
math	ACOSH	<i>Arc hyperbolic cosine.</i> Returns $\cosh^{-1} x$.
math	ASIN	<i>Arc sine.</i> Returns $\sin^{-1} x$.
math	ASINH	<i>Arc hyperbolic sine.</i> Returns $\sinh^{-1} x$.
math	ATAN	<i>Arc tangent.</i> Returns $\tan^{-1} x$.
math	ATANH	<i>Arc hyperbolic tangent.</i> Returns $\tanh^{-1} x$.
math	BASE-	<i>Base subtraction.</i> Returns the 36-bit difference of $y - x$.
math	BASE+	<i>Base addition.</i> Returns the 36-bit sum of $y + x$.
math	BASE+/ -	<i>Base change sign.</i> Returns the 36-bit 2's complement of x .
math	BASE÷	<i>Base division.</i> Returns the 36-bit quotient of $y \div x$.
math	BASE×	<i>Base multiplication.</i> Returns the 36-bit product of $y \times x$.
math	BINM	Select <i>Binary</i> mode (base 2)
math	COMPL EX	Convert two real numbers (or matrices) into a complex number (or matrix). Converts a complex number (or matrix) into two real numbers (or matrices).
math	COS	<i>Cosine.</i> Returns $\cos(x)$.
math	COSH	<i>Hyperbolic cosine.</i> Returns $\cosh(x)$.
math	CPXRES	<i>Complex-results.</i> Enable the calculator to return a complex result, even if the inputs are real numbers.
math	E↑X	<i>Natural exponential.</i> Returns e^x .
math	E↑X-1	<i>Natural exponential</i> for values of x which are close to zero. Returns $e^x - 1$, which provides a much higher accuracy in the fractional part of the result.
math	GAMM A	<i>Gamma function.</i> Returns $\Gamma(x)$.
math	INTEG	<i>Integrate</i> the selected integration program with respect to the specified variable. Parameter: variable name (indirect allowed)
math	LN	<i>Natural logarithm.</i> Returns $\ln(x)$.
math	LN1+X	<i>Natural logarithm</i> for values close to zero. Returns $\ln(1 + x)$, which provides a much higher accuracy in the fractional part of the result.
math	LOG	<i>Common logarithm.</i> Returns $\log_{10}(x)$.
math	MOD	<i>Modulo.</i> Returns the remainder for y / x .
math	MVAR	Declare a <i>menu variable</i> in a SOLVER program. Parameter: variable name.
math	N!	<i>Factorial.</i> Returns $x!$.
math	PGMIN T	Select a program to integrate. Parameter: global label (indirect allowed)
math	PGMSL V	Select a program to solve. Parameter: global label (indirect allowed) ,
math	PI	Put an approximation of π into the x-register (3.14159265359).
math	POLAR	Select <i>polar</i> coordinate mode for displaying complex numbers.
math	RAD	Select <i>Radians</i> angular mode
math	REALRE S	<i>Real-results.</i> Disables the calculator's ability to return a complex result using real-number inputs.
math	RECT	Select <i>Rectangular</i> coordinate mode for displaying complex numbers.
math	RND	<i>Round</i> the number in the x-register using the current display format.
math	SIGN	<i>Sign.</i> Return 1 for $x \geq 0$, -1 for $x < 0$, and 0 for non-numbers. Returns the unit vector of a complex number.
math	SIN	<i>Sine.</i> Returns $\sin(x)$.
math	SINH	<i>Hyperbolic sine.</i> Returns $\sinh(x)$.
math	SOLVE	<i>Solve</i> for an unknown variable. Parameter: variable name (indirect allowed)
math	SORT	<i>Square root.</i> Returns \sqrt{x} .
math	SST	<i>Single step.</i> Moves the program pointer to the next program line. (Not programmable.)
math	TAN	<i>Tangent.</i> Returns $\tan(x)$.
math	TANH	<i>Hyperbolic tangent.</i> Returns $\tanh(x)$.
math	X↑2	<i>Square.</i> Returns x^2 .
math	Y↑X	<i>Power.</i> Returns y^x .
math util	FP	Returns the <i>fractional part</i> of x .
math util	IP	Returns the <i>integer part</i> of x
matrix	←	<i>Move left</i> one element in the indexed matrix.
matrix	→	<i>Move right</i> one element in the indexed matrix.
matrix	↑	<i>Move up</i> one element in the indexed matrix.
matrix	↓	<i>Move down</i> one element in the indexed matrix.
matrix	CROSS	Returns the <i>cross product</i> of two vectors (matrices or complex numbers).
matrix	DELR	<i>Delete row.</i> Delete the current row from the indexed matrix.
matrix	DET	<i>Determinant.</i> Returns the determinant of the matrix in the x-register.
matrix	DIM	<i>Dimension</i> a matrix to x columns and y rows. If the matrix does not exist, DIM creates it. Parameter: variable name (indirect allowed)

matrix	DIM?	Returns the dimensions of the matrix in the x-register (rows to the y-register and columns to the x-register).
matrix	DOT	<i>Dot Product</i> . Returns the dot product of two vectors (matrices or complex numbers).
matrix	EDIT	<i>Edit</i> a matrix in the x-register.
matrix	EDITN	<i>Edit a named matrix</i> . Parameter: variable name (indirect allowed)
matrix	FNRM	Returns the <i>Frobenius norm</i> of the matrix in the x-register.
matrix	GETM	<i>Get matrix</i> . Copy a submatrix into the x-register from the indexed matrix.
matrix	GROW	Select <i>Grow</i> mode. Executing → or J+ causes the matrix to grow by one new row if the index pointers are at the last (lower-right) element in the matrix.
matrix	I-	<i>Decrement the row pointer</i> in the indexed matrix.
matrix	I+	<i>Increment the row pointer</i> in the indexed matrix.
matrix	INDEX	<i>Index</i> a named matrix. Parameter: variable name (indirect allowed)
matrix	INSR	<i>Insert a row</i> in the indexed matrix.
matrix	INVRT	Returns the inverse of the matrix in the x-register.
matrix	J-	<i>Decrement the column pointer</i> in the indexed matrix.
matrix	J+	<i>Increment the column pointer</i> in the indexed matrix.
matrix	NEWM AT	<i>New matrix</i> . Creates a $y \times x$ matrix in the x-register.
matrix	OLD	Recall the current element from the indexed matrix. (Equivalent to RCLEL.)
matrix	PUTM	<i>Put matrix</i> . Stores the matrix in the X-register into the indexed matrix beginning at the current element.
matrix	R<>R	<i>Row swap row</i> . Swaps the elements in rows x and y in the indexed matrix.
matrix	RCLEL	<i>Recall element</i> . Recalls the current matrix element from the indexed matrix.
matrix	RCLIJ	Recall the row- and column-pointer values (I and J) for the indexed matrix.
matrix	RNRM	Return the <i>row norm</i> of the matrix in the x-register.
matrix	RSUM	Return the <i>row sum</i> of each row of the matrix in the x-register and returns the sums in a column matrix.
matrix	STOEL	<i>Store element</i> . Stores a copy of x into the current element of the indexed matrix.
matrix	STOIJ	Moves the row- and column-pointers to $I = x$ and $J = y$ in the indexed matrix.
matrix	TRANS	Return the <i>transpose</i> of the matrix in the x-register.
matrix	UVEC	<i>Unit vector</i> . Return the unit vector for the matrix or complex number in the x-register.
matrix	WRAP	Select <i>Wrap</i> mode, which prevents the indexed matrix from growing.