

C47 Keyboard listing per layer

Layer	Row	Column	Key	Layoutlabel	Fullname	Extended description	Type	Catalog	Default	Category
unshifted	1	1	[F1]	F1	Function key 1	Function key 1	F-key			Function key
		2	[F2]	F2	Function key 2	Function key 2	F-key			Function key
		3	[F3]	F3	Function key 3	Function key 3	F-key			Function key
		4	[F4]	F4	Function key 4	Function key 4	F-key			Function key
		5	[F5]	F5	Function key 5	Function key 5	F-key			Function key
		6	[F6]	F6	Function key 6	Function key 6	F-key			Function key
	2	1	[Σ+]	Σ+	Sigma+	Enter data into the statistics matrix (STATS) (TI : nnn data point(s))	Command	Σ+		Statistics
		2	[1/x]	1/x	Reciprocal	Reciprocal (1/x) (Info : When X is a matrix 1/x inverts it ([M] ⁻¹))	Function (monadic)	1/x		Mathematics
		3	[√x]	√x	Square root	Square root	Function (monadic)	√x		Mathematics
		4	[LOG]	LOG	Common logarithm	Common logarithm (base 10)	Function (monadic)	LOG		Mathematics
		5	[LN]	LN	Natural logarithm	Natural logarithm (base e)	Function (monadic)	LN		Mathematics
		6	[XEQ]	XEQ	Execute	Execute function or program (TAM : XEQ __ TamLabel menu)	Command	XEQ		Programming
	3	1	[STO]	STO	Store (register)	Store value in register or variable ; can be followed by +, -, ×, ÷ for add into, subtract into, multiply into, divide into functions (TAM : STO __ TamStoRcl menu ; TI : Rnn: (or <var>:))	Command	STO		Stack
		2	[RCL]	RCL	Recall (register)	Recall value from register or variable can be followed by +, -, ×, ÷ for recall and add, recall and subtract, recall and multiply, recall and divide functions (TAM : RCL __ TamStoRcl menu ; TI : Rnn: (or <var>:))	Command	RCL		Stack
		3	[R↓]	R↓	Roll down	Roll down stack	Command	R↓		Stack
		4	[SIN]	SIN	Sine	Sine	Function (monadic)	SIN		Mathematics
		5	[COS]	COS	Cosine	Cosine	Function (monadic)	COS		Mathematics
		6	[TAN]	TAN	Tangent	Tangent	Function (monadic)	TAN		Mathematics
	4	1	[ENTER]	ENTER	Enter	Enter input value to X (optionally also to Y) or push/duplicate value already in X to Y	Command	ENTER↵		Stack
		2	[x↔y]	x↔y	Swap X and Y	Swap register X and register Y	Command	x↔y		Stack
		3	[CHS]	CHS	Change sign	CHange Sign	Function (monadic)	CHS		Mathematics
		4	[EEEX]	EEEX	Enter exponent	Enter EXponent (decimal input, powers of 10) (Shortcut : Equation editor to enter multiple expressions : [EEEX] enters E as a shortcut for 10 [*])	Command			Mathematics
		5	[⇐]	⇐	Backspace	Backspace (Clear input) (Moniker : BKSPC)	Command (nonpgm)			Keyboard
	5	1	[▲]	▲	BST	Scroll Up Menu (or SHOW) or Back Step	Command			Navigation
		2	[7]	7	Digit 7	Digit 7 (Code : 55)	Digit			Numeric
		3	[8]	8	Digit 8	Digit 8 (Code : 56)	Digit			Numeric
		4	[9]	9	Digit 9	Digit 9 (Code : 57)	Digit			Numeric
		5	[÷]	÷	Divide	Divide Y by X (Code : 247)	Function (dyadic)	÷		Mathematics
	6	1	[▼]	▼	Down	Scroll Down Menu (or SHOW) or Single Step	Command			Navigation
		2	[4]	4	Digit 4	Digit 4 (Code : 52)	Digit			Numeric
		3	[5]	5	Digit 5	Digit 5 (Code : 53)	Digit			Numeric
		4	[6]	6	Digit 6	Digit 6 (Code : 54)	Digit			Numeric
		5	[×]	×	Multiply	Multiply Y by X (Code : 215)	Function (dyadic)	×		Mathematics
	7	1	[f/g]	f/g	Shift f/g	Single press : shift f (yellow) ; double press : shift g (blue) (Info : SBI position depends on SShfR)	fg-shift	f/g		Keyboard
		2	[1]	1	Digit 1	Digit 1 (Code : 49)	Digit			Numeric
		3	[2]	2	Digit 2	Digit 2 (Code : 50)	Digit			Numeric
		4	[3]	3	Digit 3	Digit 3 (Code : 51)	Digit			Numeric
		5	[-]	-	Subtract	Subtract X from Y (Code : 45)	Function (dyadic)	-		Mathematics
	8	1	[EXIT]	EXIT	Exit	EXIT	Command			Keyboard
		2	[0]	0	Digit 0	Digit 0 (Code : 48)	Digit			Numeric
		3	[.]	.	Radix	Enter radix (default ".") (Info : Second press enters fraction mode)	Symbol			Numeric
		4	[R/S]	R/S	Run/Stop	Run/Stop (Program)	Command			Programming
		5	[+]	+	Add	Add X to Y ; concatenate X and Y, including numbers, dates and times + strings and vice versa (Code : 43)	Function (dyadic)	+		Mathematics
f-shift	2	1	f[Σ+]	→I	To integer	Convert to long integer/short integer (cyclic, max 1000 digits) (Info : Shortint indicated by subscript ₁₀ ; can show TI: Ovrfl<0: or Ovrfl>64bits: which can be abbreviated as OF, indicating overflow condition)	Function (cyclic ; monadic)	→I		Numeric
		2	f[1/x]	y ^x	y to the power x	Raise value in the Y-register to the power in the X-register	Function (dyadic)	y ^x		Mathematics
		3	f[√x]	x ²	Square	Square of X	Function (monadic)	x ²		Mathematics
		4	f[LOG]	10 ^x	10 to the power x	Raise 10 to the power in the X-register	Function (monadic)	10 ^x		Mathematics

Layer	Row	Column	Key	Layoutlabel	Fullname	Extended description	Type	Catalog	Default	Category
		5	f[LN]	e ^x	e to the power x	Raise e to the power in the X-register	Function (monadic)	e ^x		Mathematics
		6	f[XEQ]	[α]	Alpha input	Alpha menu is used to enter, edit and clear alpha input (Hidden : long [XEQ] ; Info : AIM = Alpha Input Mode ; opens menu α in UPPERCASE)	MENU (item)			Text
	3	1	f[STO]	x	Magnitude	Magnitude (absolute value) of complex number	Function (monadic)	x		Mathematics
		2	f[RCL]	%	Percent	X Percent of Y, keeping Y on stack	Function (dyadic)	%		Financial
		3	f[R√]	π	pi	Value of pi (Code : π)	Command			Mathematics
		4	f[SIN]	ASIN	Arc sine	Inverse sine	Function (monadic)	ASIN		Mathematics
		5	f[COS]	ACOS	Arc cosine	Inverse cosine	Function (monadic)	ACOS		Mathematics
		6	f[TAN]	ATAN	Arc tangent	Inverse tangent	Function (monadic)	ATAN		Mathematics
	4	1	f[ENTER]	COMPLEX	Complex	Convert to or from complex number (Info : a ENTER b COMPLEX returns a+bi or a±b (using angle tag, regardless of POLAR, or ADM) ; COMPLEX returns Y : a, X : b)	Function (dyadic)	COMPLEX		Mathematics
		2	f[x↔y]	LASTx	Last X	Recall last X (register L)	Command	LASTx		Stack
		3	f[CHS]	[MODE]	Mode settings	System (mode) settings with status indication and modification	MENU	MODE		Settings
		4	f[EEX]	[DISP]	Display settings	Display settings	MENU	DISP		Settings
		5	f[⇐]	↶	Undo	Restore complete stack, LASTx, STATS and system flags	Command (nonpgm)	UNDO		Keyboard
	5	1	f[▲]	≡▲	Scroll up/Backstep	Back Step	Command (nonpgm)			Programming
		2	f[7]	[EQN]	Equation	Equation editor to enter multiple expressions (Info : EIM = Equation Input Mode ; scroll through expressions using [▲] and [▼] ; constant names cannot be used as variables ; [EEX] enters E as a shortcut for 10 [^])	MENU	EQN		Equation
		3	f[8]	[ADV]	Advanced	Advanced functions	MENU	ADV		Mathematics
		4	f[9]	[MATX]	Matrix	Matrix functions (Info : A matrix is displayed in X and shown in other stack registers as [MxN Matrix] or [MxN C Matrix] for a matrix containing complex element(s) ; in edit mode, only monadic functions on matrix elements are possible, use registers to input calculation results)	MENU	MATX		Matrix
		5	f[÷]	[STAT]	Statistics	Statistics functions	MENU	STAT		Statistics
	6	1	f[▼]	≡▼	Scroll down/Single step	Single Step	Command (nonpgm)			Programming
		2	f[4]	[BASE]	Number base	Number base operations (shortint) (Split screen : Displaying X: hexadecimal ; X: shortint)	MENU	BASE		Numeric
		3	f[5]	[CONV]	Convert units	Convert units (Info : Flag CONV _{HP} (default SET) may be SET for <from> ↔ <to> unit conversion buttons ; CLEAR for <to> → ← <from> unit conversion buttons)	MENU	CONV		Unit conversion
		4	f[6]	[FLAG]	Flags	Setting, clearing and testing flags	MENU	FLAG		Programming
		5	f[x]	[PROB]	Probability	Probability functions	MENU	PROB		Probability
	7	2	f[1]	ASN	Assign	Assign function, menu or character to a keyboard key or to a button in a menu (see Ref : Assignment for options including creating user defined menus) (Info : Use button ASN ([α.] fF6) for character assignments ; this button is not shown in Program Entry Mode)	Command	ASSIGN		Text
		3	f[2]	USER	USER mode	Switch on user mode (Hidden : <none>)	Setting		OFF	Text
		4	f[3]	[P.FN]	Programming functions	User menu to quickly access selected menus and functions for programming ; all buttons are user assignable (paneled look) ; initially populated for basic programming options ; reset using PFN.R	MENU	P.FN		Programming
		5	f[-]	[PRINT]	Printing	Printing functions (Info : Print commands append data to file DATA/<YYYYMMDD-HHMMSS00>REGS.TSV in FAT ; new datafile after timeout of 2 minutes)	MENU	PRINT		Printing
	8	1	f[EXIT]	OFF	Off	Turn off calculator	Command	OFF		Keyboard
		2	f[0]	VIEW	View	View register or variable (with preceding TI) (TAM : VIEW __ Tam menu)	Command	VIEW		Stack
		3	f[.]	SHOW	Show	Show item in maximum detail, favouring register data type (tag) ; long integers up to 10 ¹²³ will be displayed in large numeric font ; up to 10 ¹¹⁶ will be displayed in the medium standard font and up to 10 ¹⁰⁰⁰ will be shown in tiny font (Info : Key [R/S] changes view for long integers (font size, next page) ; reals are also shown in SIG 6, UNIT 3, SCI 3 ; complex numbers shown in RECT, POLAR in SIG 4)	Command	SHOW		Stack
		4	f[R/S]	PRGM	Programming	Enter Program Entry Mode and activate menu P.FN (Mode : PEM = Program Entry Mode ; starts UPPERCASE)	Command			Programming
		5	f[+]	[CAT]	Catalog	Catalog of all items (functions, characters, programs, variables, menus)	MENU			Catalog
g-shift	2	1	g[Σ+]	a ^b / _c	Fraction mode	Set and cycle fraction mode : proper, improper fractions (denominator determined by setting DMX) ; or fractional approximations of irrationals ; exit mode via [.d] (g[LOG]) ; starts in mode last used or as set by flags PROPF _R and IRFRAC ; When FRCYC is SET full cycle is available, when CLEAR, flag PROPF _R is excluded (used as is) and OFF state is included in cycle (Hidden : double [.] ; Info : SBI depends on SBfrac ; /n or /max denotes maximum denominator (set by DMX) ; prefixes "<" and ">" may be shown as needed for rational fractions ; also see flag IRFRAC ; multiplication symbol according to flag MULTx ; Ref : Fractions)	Setting (cyclic ; stack)		OFF	Fractions

Layer	Row	Column	Key	Layoutlabel	Fullname	Extended description	Type	Catalog	Default	Category
		2	g[1/x]	#	Number base	Set number base ; operates on all stack registers depending on BASE _{HP} ; reset by [.d] (g[LOG]) (TAM : →INT __ TamNonReg menu ; #TAM shortcuts : B = BIN ; D = DEC ; ENTER = DEC ; H = HEX ; 0 = OCT ; Info : SBI depends on SBfrac)	Setting (pgm ; stack)	→INT		Numeric
		3	g[√x]	.ms	Minutes & seconds	Convert sexagesimal format input sequence or decimal stack value to hh:mm:ss hours or dd"mm"ss" degrees (cyclic) (Info : NIM input treated as sexagesimal (hh/dd.mmss) format ; stack input treated as decimal value)	Function (cyclic ; monadic)	.ms		Numeric
		4	g[LOG]	.d	Decimal	Convert to decimal (real) value ; clear fraction mode, base mode ; convert degrees / hours / date to real ; convert NIM input to date (according to date format set and implied conversion set by YY) ; convert complex number with zero imaginary part to real number ; in Program Entry Mode →REAL is entered (TI (degrees ; hours ; date) : decimal° ; decimal h ; yyyy-mm-dd :)	Function (monadic)			Numeric
		5	g[LN]	LBL	Label	Create local/global label (TAM : LBL __ TamLabel menu)	Command (PEM)	LBL		Programming
		6	g[XEQ]	GTO	Go to	Go to (global) label ; GTO. to go to (local) label or step ; GTO.. to move the program pointer to the end of program memory ; GTO* to go to the start of the (previous) program ; GTO+ to go to the end of the program (Hidden : longest[XEQ] ; TAM : GTO __ TamLabel menu)	Command	GTO		Programming
	3	1	g[STO]	∠	Argument (angle)	Argument (angle) of complex number	Function (monadic)	∠		Mathematics
		2	g[RCL]	Δ%	Delta percent	Delta percentage from Y to X, keeping Y on stack (TI : Δ% :)	Function (dyadic)	Δ%		Financial
		3	g[R√]	∛√	xth root	Xth root of Y	Function (dyadic)	∛√		Mathematics
		4	g[SIN]	i	Complex number (rectangular)	Enter complex number (rectangular) whether RECT is set or not ; e.g. a i b ENTER results in a + b i (Info : In NIM, works like CC with RECT set ; displayed according to flag CPX _j when in RECT mode)	Command	op_i		Mathematics
		5	g[COS]	→R	To rectangular	Transform polar to rectangular coordinates (stack conventions according to flag RP _{HP}) ; transform complex number to rectangular notation (monadic) and set RECT tag (TI : x : Re = ; y : Im = (2 stack levels))	Function (monadic ; dyadic)	→RECT		Mathematics
		6	g[TAN]	→P	To polar	Transform rectangular to polar coordinates (stack conventions according to flag RP _{HP} or ADM tag) ; transform complex number to polar notation (monadic) and set POLAR tag (TI : r = ; θ = (2 stack levels))	Function (monadic ; dyadic)	→POLAR		Mathematics
	4	1	g[ENTER]	[CPX]	Complex functions	Complex functions	MENU	CPX		Mathematics
		2	g[x↔y]	[STK]	Stack	Stack functions	MENU	STK		Stack
		3	g[CHS]	[TRG]	Trigonometry	Trigonometry and hyperbolic functions (Ref : DMS-HMS)	MENU (47)	TRG		Mathematics
		4	g[EEX]	[EXP]	Exponential	Exponential functions	MENU	EXP		Mathematics
		5	g[↔]	[CLR]	Clear	Clear flags, programs, registers, stacks, variables and reset calculator	MENU	CLR		Cleanup
	5	1	g[▲]	REGS	Register browser	Browse all registers (Shortcut : [.] : Switch register/variable view, [R/S] : Switch contents/storage view, [RCL] : Recall bottom item, [▲], [▼], A..D, I..L, 00..99 : Navigate)	Browser	REGS		Browser
		2	g[7]	[HOME]	HOME	User menu to quickly access user selected menus and functions ; all buttons are user assignable (paneled look) ; initially populated for basic scitech options ; reset using HOME.R (Hidden : triple [f/g] (HOME.3 ON) ; longer[f/g] (HOME.3 ON))	MENU	HOME		Keyboard
		3	g[8]	[FIN]	Financial	Financial calculations including time value of money (TVM)	MENU	FIN		Financial
		4	g[9]	[X.FN]	Extended functions	Extended functions (Bessel, Bernoulli, Gamma, Elliptical, Orthogonal, etc.)	MENU	X.FN		Mathematics
		5	g[÷]	[PLOT]	Plotting	Plotting and summation functions	MENU	PLOT		Plotting/sums
	6	1	g[▼]	FLGS	Flag browser	Show all flags on one page (0 = clear, 1 = set) ; show status page(s) on Up/Dn (Info : Compare FLAGS.STATUS)	Browser	FLGS		Browser
		2	g[4]	[BITS]	Bits	Bitwise operations	MENU	BITS		Numeric
		3	g[5]	[CLK]	Clock	Clock functions, including setting date and time and julian day numbers (astronomy)	MENU	CLK		Time
		4	g[6]	[REAL]	Real	Functions on real and complex numbers	MENU	REAL		Numeric
		5	g[x]	[INTS]	Integers	Short integer functions	MENU	INTS		Numeric
	7	2	g[1]	[KEYS]	Keys	Layouts, ribbons and special key assignments	MENU	KEYS		Keyboard
		3	g[2]	[α.FN]	Alpha string	Alpha (string) functions	MENU	α.FN		Text
		4	g[3]	[LOOP]	Looping	Looping (programming) functions	MENU	LOOP		Programming
		5	g[-]	[I/O]	Input/Output	Input/output functions	MENU	I/O		Input-Output
	8	1	g[EXIT]	SNAP	Screenshot	Save screenshot as bitmap ; if executed from the keyboard (g[EXIT]) also saves contents of stack or alpha buffer as text ; plays clicking sound (Hidden : (DMCP) [f/g] + [EEX] ; Info : screenshot saved in file SCREENS/<YYYYMMDD-HHMMSS00>.BMP ; data appended to file DATA/<YYYYMMDD-HHMMSS00>REGS.TSV ; new datafile after timeout of 2 minutes)	Command	SNAP		Screenshot
		2	g[0]	STOPW	Stopwatch	Stopwatch with running time and counter	App	STOPW		Time
		3	g[.]	[INFO]	Information	System information and some information about the value in the X-register	MENU	INFO		Information
		4	g[R/S]	[TEST]	Testing	Testing functions	MENU	TEST		Programming

Layer	Row	Column	Key	Layoutlabel	Fullname	Extended description	Type	Catalog	Default	Category
		5	g [+]	[CNST]	Constants	Important scientific and technical constant values (Info : Constants preceded by "# " in programs ; Type characters 1-2 to search ; TI (temporary info) is shown in description of constants)	MENU (ASM)	CNST		Catalog
alpha	2	1	alpha [Σ+]	A	A	Character A (Code : 65)	Character			Text
		2	alpha [1/x]	B	B	Character B (Code : 66)	Character			Text
		3	alpha [√x]	C	C	Character C (Code : 67)	Character			Text
		4	alpha [LOG]	D	D	Character D (Code : 68)	Character			Text
		5	alpha [LN]	E	E	Character E (Code : 69)	Character			Text
		6	alpha [XEQ]	F	F	Character F (Code : 70)	Character			Text
	3	1	alpha [STO]	G	G	Character G (Code : 71)	Character			Text
		2	alpha [RCL]	H	H	Character H (Code : 72)	Character			Text
		3	alpha [R↓]	I	I	Character I (Code : 73)	Character			Text
		4	alpha [SIN]	J	J	Character J (Code : 74)	Character			Text
		5	alpha [COS]	K	K	Character K (Code : 75)	Character			Text
		6	alpha [TAN]	L	L	Character L (Code : 76)	Character			Text
	4	1	alpha [ENTER]	ENTER	Enter	Enter input value to X (optionally also to Y) or push/duplicate value already in X to Y	Command	ENTER↵		Stack
		2	alpha [x↔y]	M	M	Character M (Code : 77)	Character			Text
		3	alpha [CHS]	N	N	Character N (Code : 78)	Character			Text
		4	alpha [EEX]	O	O	Character O (Code : 79)	Character			Text
		5	alpha [←]	↵	Backspace	Backspace (Clear input) (Moniker : BKSPC)	Command (nonpgm)			Keyboard
	5	1	alpha [▲]	↑ (HOME←)	Cursor to begin	Jump to top left of alpha input	Command			Text
		2	alpha [7]	P	P	Character P (Code : 80)	Character			Text
		3	alpha [8]	Q	Q	Character Q (Code : 81)	Character			Text
		4	alpha [9]	R	R	Character R (Code : 82)	Character			Text
		5	alpha [÷]	S	S	Character S (Code : 83)	Character			Text
	6	1	alpha [▼]	↓ (END→)	Cursor to end	Jump to bottom right of alpha input	Command			Text
		2	alpha [4]	T	T	Character T (Code : 84)	Character			Text
		3	alpha [5]	U	U	Character U (Code : 85)	Character			Text
		4	alpha [6]	V	V	Character V (Code : 86)	Character			Text
		5	alpha [x]	W	W	Character W (Code : 87)	Character			Text
	7	1	alpha [f/g]	f/g	Shift f/g	Single press : shift f (yellow) ; double press : shift g (blue) (Info : SBI position depends on SBshfr)	fg-shift	f/g		Keyboard
		2	alpha [1]	X	X	Character X (Code : 88)	Character			Text
		3	alpha [2]	Y	Y	Character Y (Code : 89)	Character			Text
		4	alpha [3]	Z	Z	Character Z (Code : 90)	Character			Text
		5	alpha [-]	_	Underscore	Character _ (Code : 95)	Character			Text
	8	1	alpha [EXIT]	EXIT	Exit	EXIT	Command			Keyboard
		2	alpha [0]	:	Colon	Character : (Hidden : alpha [0] ; Code : 58)	Character			Text
		3	alpha [.]	,	Comma	Character , (Code : 44)	Character			Text
		4	alpha [R/S]	?	Question mark	Character ? (Code : 63)	Character			Text
		5	alpha [+]	␣	Space	Character " " (Code : 32)	Character			Text
alpha f-shift	2	1	alpha f [Σ+]	a	a lowercase	Character a (Code : 97)	Character			Text
		2	alpha f [1/x]	b	b lowercase	Character b (Code : 98)	Character			Text
		3	alpha f [√x]	c	c lowercase	Character c (Code : 99)	Character			Text
		4	alpha f [LOG]	d	d lowercase	Character d (Code : 100)	Character			Text
		5	alpha f [LN]	e	e lowercase	Character e (Code : 101)	Character			Text
		6	alpha f [XEQ]	f	f lowercase	Character f (Code : 102)	Character			Text
	3	1	alpha f [STO]	g	g lowercase	Character g (Code : 103)	Character			Text
		2	alpha f [RCL]	h	h lowercase	Character h (Code : 104)	Character			Text
		3	alpha f [R↓]	i	i lowercase	Character i (Code : 105)	Character			Text
		4	alpha f [SIN]	j	j lowercase	Character j (Code : 106)	Character			Text
		5	alpha f [COS]	k	k lowercase	Character k (Code : 107)	Character			Text
		6	alpha f [TAN]	l	l lowercase	Character l (Code : 108)	Character			Text
	4	1	alpha f [ENTER]	X.EDIT	Edit X	Edit contents of X register	Command	X.EDIT		Text
		2	alpha f [x↔y]	m	m lowercase	Character m (Code : 109)	Character			Text
		3	alpha f [CHS]	n	n lowercase	Character n (Code : 110)	Character			Text
		4	alpha f [EEX]	o	o lowercase	Character o (Code : 111)	Character			Text

Layer	Row	Column	Key	Layoutlabel	Fullname	Extended description	Type	Catalog	Default	Category
		5	alpha f [⇄]	CLA	Clear alpha	Clear alphabetic input (Hidden : alpha f [⇄] ; alpha g [⇄] ; alpha long [⇄])	Command			Text
	5	1	alpha f [▲]	CASE UP	Alpha lock upwards	Move up alpha lock from a to A to N (Hidden : alpha f [▲])	Alpha-shift			Text
		2	alpha f [7]	p	p lowercase	Character p (Code : 112)	Character			Text
		3	alpha f [8]	q	q lowercase	Character q (Code : 113)	Character			Text
		4	alpha f [9]	r	r lowercase	Character r (Code : 114)	Character			Text
		5	alpha f [÷]	s	s lowercase	Character s (Code : 115)	Character			Text
	6	1	alpha f [▼]	CASE DN	Alpha lock downwards	Move down alpha lock from N to A to a (Hidden : alpha f [▼])	Alpha-shift			Text
		2	alpha f [4]	t	t lowercase	Character t (Code : 116)	Character			Text
		3	alpha f [5]	u	u lowercase	Character u (Code : 117)	Character			Text
		4	alpha f [6]	v	v lowercase	Character v (Code : 118)	Character			Text
		5	alpha f [x]	w	w lowercase	Character w (Code : 119)	Character			Text
	7	1	alpha f [f/g]	<f>lipchar	Flip case (one character)	Characters <f>lipchar (Hidden : alpha f + <char>)	Character			Text
		2	alpha f [1]	x	x lowercase	Character x (Code : 120)	Character			Text
		3	alpha f [2]	y	y lowercase	Character y (Code : 121)	Character			Text
		4	alpha f [3]	z	z lowercase	Character z (Code : 122)	Character			Text
		5	alpha f [-]	ø	Euler's e	Character ø (Hidden : alpha f [-] ; Code : 8519)	Character			Text
	8	1	alpha f [EXIT]	OFF	Off	Turn off calculator	Command	OFF		Keyboard
		2	alpha f [0]	;	Semicolon	Character ; (Hidden : alpha f [0] ; Code : 59)	Character			Text
		3	alpha f [.]	#	Number sign	Character # (Hidden : alpha f [.]; Code : 35)	Character			Text
		4	alpha f [R/S]	!	Exclamation mark	Character ! (Hidden : alpha f [R/S] ; Code : 33)	Character			Text
		5	alpha f [+]	=	Equal	Character = (Hidden : alpha f [+]; Code : 61)	Character			Text
alpha g-shift	2	1	alpha g [Σ+]	Σ	SIGMA	Character Σ (Hidden : alpha g [Σ+]; Code : 931)	Character			Text
		2	alpha g [1/x]	^	Circumflex accent	Character ^ (Hidden : alpha g [1/x]; Code : 94)	Character			Text
		3	alpha g [√x]	√	Square root	Character √ (Hidden : alpha g [√x]; Code : 8730)	Character			Text
		4	alpha g [LOG]	LOG	Common logarithm (string)	Characters LOG (Hidden : alpha g [LOG] ("LOG"))	Character			Text
		5	alpha g [LN]	LN	Natural logarithm (string)	Characters LN (Hidden : alpha g [LN] ("LN"))	Character			Text
		6	alpha g [XEQ]	α	alpha lowercase	Character α (Hidden : alpha g [XEQ]; Code : 945)	Character			Text
	3	1	alpha g [STO]		Bar	Character (Hidden : alpha g [STO]; Code : 124)	Character			Text
		2	alpha g [RCL]	Δ	DELTA	Character Δ (Hidden : alpha g [RCL]; Code : 916)	Character			Text
		3	alpha g [R√]	π	pi lowercase	Character π (Hidden : alpha g [R√]; Code : 960)	Character			Text
		4	alpha g [SIN]	SIN	Sine (string)	Characters SIN (Hidden : alpha g [SIN] ("SIN"))	Character			Text
		5	alpha g [COS]	COS	Cosine (string)	Characters COS (Hidden : alpha g [COS] ("COS"))	Character			Text
		6	alpha g [TAN]	TAN	Tangent (string)	Characters TAN (Hidden : alpha g [TAN] ("TAN"))	Character			Text
	4	1	alpha g [ENTER]	↵	Carriage return	Character ↵ (Code : 8629)	Character			Text
		2	alpha g [x↔y]	↔	Right over left arrow	Character ↔ (Hidden : alpha g [x↔y]; Code : 8644)	Character			Text
		3	alpha g [CHS]	±	Plus-minus	Character ± (Hidden : alpha g [CHS]; Code : 177)	Character			Text
		4	alpha g [EEX]	<E>	Exponent sign (AIM)	Character E (displays as outline E in numeric font) (Hidden : alpha g [EEX]; Code : 8307)	Character			Text
		5	alpha g [⇄]	CLA	Clear alpha	Clear alphabetic input (Hidden : alpha f [⇄] ; alpha g [⇄] ; alpha long [⇄])	Command			Text
	5	1	alpha g [▲]	↑ (α ^{SUP})	Superscript	Superscript (Hidden : alpha g [▲])	Alpha-shift			Text
		2	alpha g [7]	7	Digit 7	Character 7 (Code : 55)	Character			Numeric
		3	alpha g [8]	8	Digit 8	Character 8 (Code : 56)	Character			Numeric
		4	alpha g [9]	9	Digit 9	Character 9 (Code : 57)	Character			Numeric
		5	alpha g [÷]	÷	Obelus	Character ÷ (Code : 247)	Character			Text
	6	1	alpha g [▼]	↓ (α _{SUB})	Subscript	Subscript (Hidden : alpha g [▼])	Alpha-shift			Text
		2	alpha g [4]	4	Digit 4	Character 4 (Code : 52)	Character			Numeric
		3	alpha g [5]	5	Digit 5	Character 5 (Code : 53)	Character			Numeric
		4	alpha g [6]	6	Digit 6	Character 6 (Code : 54)	Character			Numeric
		5	alpha g [x]	×	Cross	Character × (Code : 215)	Character			Text
	7	1	alpha g [f/g]	di<g>it	Set numeric (one digit)	Characters di<g>it (Hidden : alpha g + <char>)	Character			Text
		2	alpha g [1]	1	Digit 1	Character 1 (Code : 49)	Character			Numeric
		3	alpha g [2]	2	Digit 2	Character 2 (Code : 50)	Character			Numeric
		4	alpha g [3]	3	Digit 3	Character 3 (Code : 51)	Character			Numeric
		5	alpha g [-]	-	Minus	Minus (Code : 45)	Character			Text
	8	1	alpha g [EXIT]	SNAP	Screenshot	Save screenshot as bitmap ; if executed from the keyboard (g[EXIT]) also saves contents of stack or alpha buffer as text ; plays clicking sound (Hidden : (DMCP) [f/g] + [EEX] ; Info : screenshot saved in file SCREENS/<YYYYMMDD-HHMMSS00>.BMP ; data appended to file DATA/<YYYYMMDD-HHMMSS00>REGS.TSV ; new datafile after timeout of 2 minutes)	Command	SNAP		Screenshot

Layer	Row	Column	Key	Layoutlabel	Fullname	Extended description	Type	Catalog	Default	Category
		2	alpha g [0]	0	Digit 0	Character 0 (Code : 48)	Character			Numeric
		3	alpha g [.]	.	Full stop	Character . (Code : 46)	Character			Text
		4	alpha g [R/S]	/	Slash	Character / (Code : 47)	Character			Text
		5	alpha g [+]	+	Plus	Plus (Code : 43)	Character			Text