

FullName	Type	Label	Catalog	Description
0	Digit	[0]		Digit 0 (48)
1	Button (black)	1		Soft button 1 for selection by KEYG or KEYX function
1	Digit	[1]		Digit 1 (49)
2	Button (black)	2		Soft button 2 for selection by KEYG or KEYX function
2	Digit	[2]		Digit 2 (50)
3	Button (black)	3		Soft button 3 for selection by KEYG or KEYX function
3	Digit	[3]		Digit 3 (51)
4	Button (black)	4		Soft button 4 for selection by KEYG or KEYX function
4	Digit	[4]		Digit 4 (52)
5	Button (black)	5		Soft button 5 for selection by KEYG or KEYX function
5	Digit	[5]		Digit 5 (53)
6	Button (black)	6		Soft button 6 for selection by KEYG or KEYX function
6	Digit	[6]		Digit 6 (54)
7	Button (black)	7		Soft button 7 for selection by KEYG or KEYX function
7	Digit	[7]		Digit 7 (55)
8	Button (black)	8		Soft button 8 for selection by KEYG or KEYX function
8	Digit	[8]		Digit 8 (56)
9	Button (black)	9		Soft button 9 for selection by KEYG or KEYX function
9	Digit	[9]		Digit 9 (57)
10	Button (black)	10		Soft button 10 for selection by KEYG or KEYX function
11	Button (black)	11		Soft button 11 for selection by KEYG or KEYX function
12	Button (black)	12		Soft button 12 for selection by KEYG or KEYX function
13	Button (black)	13		Soft button 13 for selection by KEYG or KEYX function
14	Button (black)	14		Soft button 14 for selection by KEYG or KEYX function
15	Button (black)	15		Soft button 15 for selection by KEYG or KEYX function
16	Button (black)	16		Soft button 16 for selection by KEYG or KEYX function
17	Button (black)	17		Soft button 17 for selection by KEYG or KEYX function
18	Button (black)	18		Soft button 18 for selection by KEYG or KEYX function
SUN	Character	☉		Character ☉ (8857)
,	Character	,		Character , (44)
.	Symbol	[.]		Enter radix (default ".") (Second press enters fraction mode)
...EL	Command (TAM)	...EL		Presented in TAM menus for commands accessing matrix element
...IJ	Command (TAM)	...IJ		Presented in TAM menus for commands accessing matrix index
(-1)*	Function (monadic)	(-1)*	(-1)*	Cosine of πx [+ i * sine of πx for complex parameter] (Esp. for non-integer x)
(Data) entry?	Command	ENTRY?	ENTRY?	Test internal entry flag for data entry
(Plot) Central	Command (special)	CENTRL	CENTRL	Orthogonal curve fit and show parameters, requires 30 data pairs (Opens the graph mode and (re)starts menu SCATR)
$(\chi^2)^{-1}$	Function (tbd)	$(\chi^2)^{-1}$	$(\chi^2)^{-1}$	χ^2 probability inverse function
°C to °F	Function (linked ; monadic)	°C→°F		Convert degrees Celsius to degrees Fahrenheit (°F:)
°F to °C	Function (linked ; monadic)	°F→°C		Convert degrees Fahrenheit to degrees Celsius (°C:)
$\sqrt{(1+x^2)}$	Function (legacy ; monadic)	$\sqrt{(1+x^2)}$	$\sqrt{(1+x^2)}$	$\sqrt{(1+x^2)}$
1's complement	Setting (pgm)	1COMPL	1COMPL ()	Set 1's complement mode for shortint (SBI depends on SBint)
10 to the power x	Function (monadic)	10 ^x	10 ^x	Raise 10 to the power in the X-register
10s complement	Setting (pgm)	10CMPL		Set BCD 10s complement mode for shortint
12 hour clock	Setting (pgm)	CLK12	CLK12 ()	12 hour time display
1st derivative	Function (tbd)	f' here		First derivative (f' =)
1st eccentricity squared	Constant (#47)	Se ²		sq.eccent1 Se ² = +6.69437999014 × 10 ⁻³
1st radiation constant	Constant (#5)	c ₁		c.radiatn1 c ₁ = +3,741771852192758011367155555929985 × 10 ⁻¹⁶ (Wm ²)
2 to the power x	Function (monadic)	2 ^x	2 ^x	Raise 2 to the power in the X-register
2's complement	Setting (pgm)	2COMPL	2COMPL (•)	Set 2's complement mode for shortint (SBI depends on SBint)

FullName	Type	Label	Catalog	Description
24 hour clock	Setting (pgm)	CLK24	CLK24 (*)	24 hour time display
2nd derivative	Function (tbd)	f''here		Second derivative (f'' =)
2nd eccentricity squared	Constant (#48)	Se' ^{1,2}		sq.eccent2 Se' ^{1,2} = +6.73949674228 × 10 ⁻³
2nd radiation constant	Constant (#6)	C ₂		c.radiatn2 c ₂ = +1,438776877503933802146671601543912 × 10 ⁻² (mK)
9s complement	Setting (pgm)	9CMPL		Set BCD 9s complement mode for shortint
A	Character	Á		Character A (65)
A acute	Character	Á		International character Á (193)
a acute lowercase	Character	á		International character á (225)
A breve	Character	Ă		International character Ă (258)
a breve lowercase	Character	ă		International character ă (259)
A circumflex	Character	Â		International character Â (194)
a circumflex lowercase	Character	â		International character â (226)
A diaeresis	Character	Ä		International character Ä (196)
a diaeresis lowercase	Character	ä		International character ä (228)
A grave	Character	À		International character À (192)
a grave lowercase	Character	à		International character à (224)
a lowercase	Character	a		Character a (97)
A macron	Character	Ā		International character Ā (256)
a macron lowercase	Character	ā		International character ā (257)
A ogonek	Character	Ą		International character ą (260)
a ogonek lowercase	Character	ą		International character ą (261)
A ring	Character	Å		International character Å (197)
a ring lowercase	Character	å		International character å (229)
A tilde	Character	Ã		International character Ã (195)
a tilde lowercase	Character	ã		International character ã (227)
A-F keys	Setting	KEY _{A-F}	KEY _{A-F} [.]	Set row 2 keys to be used for entry of digits A-F while in #BASE mode
Absolute	Function (Legacy)	ABS	ABS	Magnitude (absolute value) of complex number
Access blue functions	MENU	BLUE47	BLUE47	Access all C47 gShifted functions and a few fShifted, supporting layout DM42 (Assigned to SETUP (fShifted [0]) in layout DM42)
Access DMCP menu	Command	DMCP	DMCP	Access DMCP menu (use EXIT to return without reset ; hardware only) (Does not autosave backup file to FAT)
Accuracy	Variable (real)	ACC	ACC	Accuracy of integration (reserved real variable) (ACC :)
acre to ha	Function (linked ; monadic)	acre→ha		Convert acre to hectare (ha:)
acre _{US} to ha	Function (linked ; monadic)	acre _{US} → ha		Convert US acre to hectare (ha:)
Activate alpha (TAM)	Shortcut (TAM)	α (TAM)		TAM shortcut for activating alpha input (Shortcut α (C47.26.41))
Activate USB disk	Command	ActUSB	ActUSB	Activate USB disk without exiting to DMCP (hardware only) (Does not autosave backup file to FAT)
Add	Function (dyadic)	[+]	+	Add X to Y ; concatenate X and Y (Concatenation includes numbers, dates and times + strings and vice versa)
Add differential curve	Setting	Δy/Δx		Add another curve, the graphical point-to-point differential of the main graph
Add integral curve	Setting	ΣyΔx		Add another curve, the graphical integral calculated from the point-to-point discrete integral (Vertical placement of the integral is at y = RMS(y))
Add into	Function (monadic)	ST0+	ST0+	Add X to register or variable (ST0+ __ TamStoRcl(Alpha) menu)
Add margin to cost	Function (dyadic)	%+MG	%+MG	Add margin of X to cost of Y
Add RMS curve	Setting	RMS		Add another curve, the graphical root mean square, accumulated from left to right.
Add timestamp	Command	ADD		Add timestamp to statistics matrix (STATS)
Advanced	MENU	ADV	ADV	Advanced functions
AE	Character	Æ		International character Æ (198)
ae lowercase	Character	æ		International character æ (230)

FullName	Type	Label	Catalog	Description
All digits notation	Setting (pgm)	ALL	ALL ₃ (•)	Set numeric display mode to ALL digits notation ; display all digits as far as possible with max. nn decimal zeros (ALL __ TamNonReg menu)
ALL/FIX ENG(ineering)	Flag	ALLENG		Real number exceeding displayable range will be shown as either SCI (if clear) or ENG (if set) (FF A)
Allocate local registers	Command (PEM)	LocR	LocR	Allocates n local registers and 16 local flags (current routine) (LocR __ TamNonReg menu)
Almost equal	Character	≈		Character ≈ (8776)
Alpha	Character	Α		Greek character Α (913)
Alpha capslock	Flag (system)	αCAP		Set for capital letters, clear for lower case
Alpha dot	MENU	α•	α•	Special characters
Alpha Entry	Flag	ALP.IN		Alpha entry active
Alpha graphics	Command	AGRAPH	AGRAPH	Show a 64 column pattern according to variable GRAMOD (OR: 0 ; SET: 1 ; OFF: 2 ; XOR: 3) (AGRAPH __ Tam menu)
Alpha input	MENU (item)	α		Alpha menu is used to enter, edit and clear alpha input (AIM = Alpha Input Mode ; opens menu α in UPPERCASE)
Alpha input	MENU	ALPHA	ALPHA	Alpha menu is used to enter, edit and clear alpha input (Double [XEQ] ; Info : Use ASN in this menu to assign to Mya ; CAPS also toggles initial case of Greek and international menus)
Alpha input mode	Flag	ALPHA		Set for AIM, else clear
Alpha international	MENU (ASM)	αINTL	αINTL	International characters (Lowercase menu via [▼] ; Type characters 1-2 to search)
alpha international	MENU (ASM)	αintl lower	αintl	Lowercase international characters (Uppercase menu via [▲] ; Type characters 1-2 to search)
Alpha length	Function (monadic)	αLENG?	αLENG?	String length (αLENG? __ Tam menu)
Alpha lock downwards	Command	αLock↓		Move down alpha lock from N to A to a
Alpha lock upwards	Command	αLock↑		Move up alpha lock from a to A to N
alpha lowercase	Character	α		Greek character α (945)
Alpha Math	MENU	αMATH	αMATH	Mathematical symbols
Alpha parse	Command (strike)	αPARSE	αPARSE	Parse alpha input for numeric content (in development)
Alpha position	Function (monadic)	αPOS?	αPOS?	Position in string in variable or stack of substring in X (starting from 0) (αPOS? __ Tam menu)
Alpha rotate left	Function (monadic)	αRL	αRL	Alpha rotate left (αRL __ Tam menu)
Alpha rotate right	Function (monadic)	αRR	αRR	Alpha rotate right (αRR __ Tam menu)
Alpha shift left	Function (monadic)	αSL	αSL	Alpha shift left (αSL __ Tam menu)
Alpha shift right	Function (monadic)	αSR	αSR	Alpha shift right (αSR __ Tam menu)
Alpha string	MENU	α.FN	α.FN	Alpha (string) functions
Alpha to X	Function (monadic)	α→x	α→x	Convert first alpha character of string in variable or register to character code (hexadecimal) (α→x __ Tam menu)
alpha tonos lowercase	Character	ά		Greek character ά (940)
ALPHA..OMEGA	MENU	A..Ω	A..Ω	Uppercase Greek characters (Lowercase menu via [▼])
alpha..omega	MENU	α..ω lower	α..ω	Lowercase Greek characters (Uppercase menu via [▲])
Ampersand	Character	&		Character & (38)
And	Character	∧		Character ∧ (8743)
AND	Function (dyadic)	AND	AND	Logical AND (bitwise)
Angle conversion	MENU	Angle:	Angle:	Convert between units of angle
Angle variables	MENU	ANGLES	ANGLES	Auto-generated catalog of variables of the specified type: angle
Angular display mode	Variable (longint)	ADM	ADM	Angular display mode (system real variable, write protected) (RAD: 2 ; ; GRAD: 4 ; DEG: 0 ; [D.MS: 1 ; MULπ: 3])
Annual interest rate	Variable (real)	i%/a	i%/a	Annual interest rate (reserved real variable)
Annual periods	Variable (real)	PER/a	PER/a	Annual number of payment or compounding periods for loan or investment
Anti-clockwise vectors	Setting	VECT		PLSTAT only: treat coordinate pairs as vectors, anti-clockwise from the positive x-axis as reference
Arc cosine	Function (legacy)	ARCCOS	ARCCOS	Inverse cosine
Arc cosine	Function (monadic)	ACOS	ACOS	Inverse cosine

FullName	Type	Label	Catalog	Description
Arc cosine	Function (monadic)	ACOS		Inverse cosine
Arc sine	Function (monadic)	ASIN	ASIN	Inverse sine
Arc sine	Function (legacy)	ARCSIN	ARCSIN	Inverse sine
Arc sine	Function (monadic)	ASIN		Inverse sine
Arc tangent	Function (monadic)	ATAN	ATAN	Inverse tangent
Arc tangent	Function (legacy)	ARCTAN	ARCTAN	Inverse tangent
Arc tangent	Function (monadic)	ATAN		Inverse tangent
Area conversion	MENU	Area:	Area:	Convert between units of area
Argument (angle)	Function (monadic)	∠	∠	Argument (angle) of complex number
Arithmetic Geometric Mean	Function (dyadic)	AGM	AGM	Arithmetic geometric mean of X and Y
Arithmetic means	Command	\bar{x}	\bar{x}	Arithmetic means (\bar{x} ; \bar{y} = (2 stack levels))
Arithmetic shift right	Function (monadic)	ASR	ASR	Arithmetic shift right by n (ASR ___ TamNonReg menu)
Assess	MENU (item)	ASSESS		Assess curve fitting
Assess	MENU	ASSESS		Assess curve fitting
Assign	Command	ASN	ASSIGN	1. Assign function or menu to key (in USER mode) using ASN (fShifted [1]) + <function>/<menu> + <key> on the keyboard 2. assign character to soft button using ASN (on menu α) + <character> + <soft button> on menu Myα 3. assign function or menu to soft button using ASN (fShifted [1]) + <function>/<menu> + <soft button> on menu MyMenu or user defined menu (only functions can be assigned in the case of a user defined menu and it must be active already) 4. create user defined menu using ASN (fShifted [1]) + USER (fShifted [2]) + <menu-name> (this menu will appear in the menu catalog and can be deleted using DELITM) (Select function from keyboard, catalog or enter name manually ; select character from keyboard or any of the alpha menus ; select menu from catalog or enter name manually ; unassign using ASN + ENTER)
Asterisk	Character	*		Character * (42)
At	Character	@		Character @ (64)
atm to Pa	Function (linked ; monadic)	atm→Pa		Convert atmosphere to Pascal (Pa:)
Atomic mass constant	Constant (#30)	m_u		mass.atom $m_u = +1.6605390666 \times 10^{-27}$ (kg)
au to m	Function (linked ; monadic)	au→m		Convert astronomical unit to meter (m:)
Auto Execute (Program)	Flag	AUTXEQ		Auto-execute program when turning on calculator
Auto Off (Calculator)	Flag	AUTOFF		Automatic shutdown active (10 m)
Automatic Stack Lift	Flag (system)	ASLIFT		Automatic stack lift active (SBI is SIM only)
Autoscale x-axis	Setting	X.AXIS		PLSTAT only: autoscale Y so that the x-axis becomes visible
Autoscale y-axis	Setting	Y.AXIS		PLSTAT only: autoscale X so that the x-axis becomes visible
Avogadro's number	Constant (#35)	N_A		nr.avogadro $N_A = +6.02214076 \times 10^{23}$ (/mol)
B	Character	B		Character B (66)
b lowercase	Character	b		Character b (98)
B_n	Function (monadic)	B_n	B_n	Bernoulli number (new definition)
B_n^*	Function (monadic)	B_n^*	B_n^*	Bernoulli number (old definition)
Back	Command (PEM)	BACK	BACK	Jumps n steps back in program (BACK ___ TamNonReg menu)
Back slash	Character	\		Character \ (92)
Backspace	Command (nonpgm)	↵		Backspace (Clear NIM) (BKSPC)
Bar	Symbol			Open/close absolute value (124)
bar to Pa	Function (linked ; monadic)	bar→Pa		Convert bar to Pascal (Pa:)
barrel to m ³	Function (linked ; monadic)	barrel→m ³		Convert barrel to cubic meter (m ³ :)
Battery	Command	BATT?	BATT?	Battery voltage ±1 mV
Battery	Character	🔋		Character 🔋 (9258)
BCD unsigned	Setting (pgm)	BCDUNS		Set BCD unsigned mode for shortint
Beep	Command	BEEP	BEEP	Play beep sound (4 tones)

FullName	Type	Label	Catalog	Description
Best Fit	Command	BestF	BestF	Select best curve fit model (BestF ___ TamNonReg menu)
Best Fit Setting	Command	BestF?	BestF?	User setting for which models are allowed in the LR analysis ; set in menu MODEL
Beta	Character	B		Greek character B (914)
beta lowercase	Character	β		Greek character β (946)
Binary	Setting (pgm)	BIN	BIN	Convert X to binary and toggle binary mode
Binary coded decimal	Setting	BCD	BCD []	Set BCD mode for shortint
Binary logarithm	Function (monadic)	LB	LB	Binary logarithm (base 2)
Binomial (inverse)	Function (tbd)	Binom ⁻¹	Binom ⁻¹	Binomial probability inverse function
Binomial cdf (lower)	Function (tbd)	Binom _▲	Binom _▲	Binomial cumulative distribution (lower tail)
Binomial cdf (upper)	Function (tbd)	Binom _▲	Binom _▲	Binomial cumulative distribution (upper tail)
Binomial distribution	MENU	Binom:	Binom:	Binomial probability distribution (RegI = p = probability ; RegJ = n = number of samples)
Binomial pdf	Function (tbd)	Binom _p	Binom _p	Binomial probability density function
Bit clear?	Function (monadic)	BC?	BC?	Test if bit n is clear (BC? ___ TamNonReg menu)
Bit set?	Function (monadic)	BS?	BS?	Test if bit n is set (BS? ___ TamNonReg menu)
Bits	MENU	BITS	BITS	Bitwise operations
Bohr radius	Constant (#1)	a ₀		rad.bohr a ₀ = +5.29177210903 × 10 ⁻¹¹ (m)
Bohr's magneton	Constant (#65)	μ _B		magn.both μ _B = +9.274010078 × 10 ⁻²⁴ (J/T)
Boltzmann constant	Constant (#20)	k		c.boltzmn k = +1.380649 × 10 ⁻²³ (J/K)
Box markers	Command	BOX		Use boxes as markers for the main graph
brds to in.	Function (linked ; monadic)	brds→in.		Convert beardsecond to inch (in.:
brds to m	Function (linked ; monadic)	brds→m		Convert beardsecond to meter (m.:
BST	Command	[▲]		Scroll Up Menu (or SHOW) or Back Step
Btu to J	Function (linked ; monadic)	Btu→J		Convert British Thermal Unit to Joule (J.:
Bullet	Character	•		Character • (8729)
Busy	SBI	⏸		Calculator busy
Buzzer	Command	BUZZ	BUZZ	Play sound (input from stack : frequency in Hz (0 = silent) ; duration in ms (max 2000))
Byte swap	Function (monadic)	B.SWP	B.SWP	Swap bytes
C	Character	C		Character C (67)
C acute	Character	Ć		International character Ć (262)
c acute lowercase	Character	ć		International character ć (263)
C caron	Character	Č		International character Č (268)
c caron lowercase	Character	č		International character č (269)
C cedilla	Character	Ç		International character Ç (199)
c cedilla lowercase	Character	ç		International character ç (231)
c lowercase	Character	c		Character c (99)
C47 keyboard layout	Layout	C47		C47: Final classic single shift, replaces C43 (DM42 mould)
cal to J	Function (linked ; monadic)	cal→J		Convert calorie to Joule (J.:
Calculate	Command (nonpgm)	Calc		Calculate result for equation
Capslock	Setting	CAPS		Lock uppercase alpha input (Toggling clears Numlock)
carat to g	Function (linked ; monadic)	carat→g		Convert carat to gram (g.:
Carry	Flag	CARRY		Status of carry bit (FF C ; Info : SBI depends on SBoc)
Case	Command (PEM)	CASE	CASE	Case statement, skips number of steps (CASE normally followed by number of GT0's) (CASE ___ Tam menu)
Catalan's constant	Constant (#14)	G _c		c.catalan G _c = +9,159655941772190150546035149323841 × 10 ⁻¹
Catalog	MENU	CAT		Catalog of all items (functions, characters, programs, variables, menus)
Cauchy (inverse)	Function (tbd)	Cauch ⁻¹	Cauch ⁻¹	Cauchy probability inverse function
Cauchy cdf (lower)	Function (tbd)	Cauch _▲	Cauch _▲	Cauchy cumulative distribution (lower tail)
Cauchy cdf (upper)	Function (tbd)	Cauch _▲	Cauch _▲	Cauchy cumulative distribution (upper tail)
Cauchy fit	Setting	CauchF	CauchF []	Cauchy curve fitting
Cauchy pdf	Function (tbd)	Cauch _p	Cauch _p	Cauchy probability density function

FullName	Type	Label	Catalog	Description
Cauchy-Lorentz distribution	MENU	Cauch:	Cauch:	Cauchy-Lorentz probability distribution (RegI = x_0 = location ; RegJ = γ = scale)
Ceiling	Function (monadic)	ceil	ceil	Ceiling (type real)
Change sign	Function (monadic)	+/-	CHS	Change sign
Characteristic impedance of vacuum	Constant (#54)	Z_0		imped.vac $Z_0 = +3,767303134617706554681984004203193 \times 10^2$ (Ω)
Characters	MENU	CHARS	CHARS	Access to all character submenus (international, greek, math, MyAlpha, alphaDot)
Chebyshev polynomials (1st)	Function (dyadic)	T_n	T_n	Chebyshev polynomials of the 1st kind ($x = X$; $n = Y$)
Chebyshev polynomials (2nd)	Function (dyadic)	U_n	U_n	Chebyshev polynomials of the 2nd kind ($x = X$; $n = Y$)
Check mark	Character	✓		Character ✓ (10003)
Chi	Character	Χ		Greek character Χ (935)
chi lowercase	Character	χ		Greek character χ (967)
chī to m	Function (linked ; monadic)	chī→m		Convert chī to meter (m)
Chinese formatting	Setting (pgm)	CHINA	SETCHN	Set to Chinese regional formats (date, time, calendar, number formatting) (First Gregorian day set: 1949-10-01)
Classic Rect/Polar	Setting	RP _{HP}	RP _{HP} [*]	Set for the classic (HP) stack conventions for →RECT and →POLAR ; Clear to follow C47 conventions of CC, COMPLEX and <i>i</i> (swapped) (Classic means $X = x$ (Re) , $Y = y$ (Im) for RECT ; $X = r$, $Y = \theta$ for POLAR)
Classic Rect/Polar	Flag	RP _{HP}		Set for the classic (HP) stack conventions for →RECT and →POLAR ; Clear to follow C47 conventions of CC, COMPLEX and <i>i</i> (swapped) (Classic means $X = x$ (Re) , $Y = y$ (Im) for RECT ; $X = r$, $Y = \theta$ for POLAR)
Classical electron radius	Constant (#39)	r_e		rad.elec $r_e = +2.8179403262 \times 10^{-15}$ (m)
Clear	MENU	CLR	CLR	Clear flags, programs, registers, stacks, variables and reset calculator
Clear (current) program	Command	CLP	CLP	Clear current program ; careful! (CLP _ TamLbl(Alpha) menu)
Clear (delete) Backup	Command	CLBKUP	CLBKUP	Delete configuration backup file (LOAD ; SAVE)
Clear (program) menu	Command	CLMENU	CLMENU	Clear the programmable menu
Clear (user) flags	Command	CLFALL	CLFALL	Clear all user flags
Clear all	Command	CLALL	CLALL	Clear all (programs and data)
Clear all programs	Command	CLPALL	CLPALL	Clear all programs
Clear all registers	Command	CLREGS	CLREGS	Clear all registers
Clear alpha	Command	CLA		Clear alphabetic input (Longpress AIM [↔])
Clear bit	Function (monadic)	CB	CB	Clear bit n (CB __ TamNonReg menu)
Clear current (program) variables	Command	CLCVAR	CLCVAR	Clear current program variables
Clear flag	Command	CF	CF	Clear flag (CF __ TamFlag menu)
Clear graph	Command	CLGRF	CLGRF	Clear graph
Clear LCD (screen)	Command	CLLCD	CLLCD	Clear (part of) the display, depending on X and Y
Clear modes	Command	CLRMOD	CLRMOD	Clear all menus, entry, fraction and base modes (Longerpress (also in AIM) [EXIT])
Clear number	Command	CLN		Clear numeric input (Longpress NIM [↔])
Clear registers	Command	R-CLR	R-CLR	Clear registers (sss.nn means clear registers from sss through sss + nn - 1)
Clear stack	Command	CLSTK	CLSTK	Clear all stack data (Longpress [↔])
Clear statistics	Command	CLΣ	CLΣ	Clear all statistics data (and delete STATS and HISTO matrices)
Clear X	Command	CLX	CLX	Clear X-register
Clock	Character	🕒		Character 🕒 (8986)
Clock	MENU	CLK	CLK	Clock functions, including setting date and time and julian day numbers (astronomy)
Clockwise vectors	Setting	N.VECT		PLSTAT only: treat coordinate pairs as navigation vectors, clockwise from the positive y-axis as reference
Colon equals	Character	:=		Character := (8788)
Combinations	Function (dyadic)	C_{yx}	COMB	Combinations of X out of Y
Common logarithm	Function (monadic)	LOG		Common logarithm (base 10)
Common logarithm	Function (monadic)	LOG	LOG	Common logarithm (base 10)
Complementary error function	Function (monadic)	erfc	erfc	Complementary error function
Complete elliptic integral (1st)	Function (monadic)	K(m)	K(m)	Complete elliptic integral of the 1st kind ($m = X$)

FullName	Type	Label	Catalog	Description
Complete elliptic integral (2nd)	Function (monadic)	E(m)	E(m)	Complete elliptic integral of the 2nd kind (m = X)
Complete elliptic integral (2rd)	Function (dyadic)	$\Pi(n,m)$	$\Pi(n,m)$	Complete elliptic integral of the 3rd kind (n = X ; m = Y)
Complex	Function (dyadic)	COMPLEX	COMPLEX	Convert to or from complex number (a ENTER b COMPLEX returns a+bi or a \angle b (using b angle tag or ADM) ; COMPLEX returns Y : a, X : b)
Complex	MENU	CPX	CPX	Complex functions
Complex C	Character	C		Character C (8450)
Complex j	Flag	CPXj		Set for the letter j representing the imaginary number (i or j displayed in stack and on soft buttons)
Complex multiplier	Setting	CPXmul	CPXmul []	Complex multiplier (If clear, trailing i or j is used)
Complex results	Setting	CPXRES	CPXRES [*]	Set to allow complex results for real input ; auto-set when entering complex input (FF I ; Info : SBI depends on SBcr)
Complex results	Flag	CPXRES		Set to allow complex results for real input ; auto-set when entering complex input (FF I ; Info : SBI depends on SBcr)
Complex results	Command	CPXRES1	CPXRES1	Allow complex results for real input ; an error will not occur for such events (For programming purposes)
Complex solver	Command (nonpgm)	cpxSlv		Complex solver for equation, with iteration counter, interrupt by keypress ; tolerance set by SDIGS
Complex to real	Function (monadic)	CX \rightarrow RE	CX \rightarrow RE	Convert complex to reals (in POLAR, using angle tag or ADM) (Re = ; Im = or r = ; θ = (2 stack levels))
Complex variables	MENU	CPXS	CPXS	Auto-generated catalog of variables of the specified type: complex
Complex?	Function (monadic)	CPX?	CPX?	Test X is complex
Compose 3x1 matrix	Function (triadic)	zyx \rightarrow M	zyx \rightarrow M	Create 3x1 matrix from ZYX (M \rightarrow zyx)
Compose-cut	Command (nonpgm)	CC		Complex closing, composing, cutting, and converting (a CC b ENTER returns a+bi or a \angle b (using b angle tag or ADM) ; CC returns Y : a, X : b)
Compton wavelength of the electron	Constant (#61)	λ_c		wavln.elec $\lambda_c = +2.42631023867 \times 10^{-12}$ (m)
Compton wavelength of the neutron	Constant (#62)	λ_{cn}		wavln.neu $\lambda_{cn} = +1.31959090581 \times 10^{-15}$ (m)
Compton wavelength of the proton	Constant (#63)	λ_{cp}		wavln.prot $\lambda_{cp} = +1.32140985539 \times 10^{-15}$ (m)
Conductance quantum	Constant (#13)	G_0		cond.quant $G_0 = +7.748091729863650646680823323308764 \times 10^{-5}$ (Ω^{-1})
Configuration	MENU (item)	CFG		Activates menu for setting system flags using FF (Flip flag) function (SYS.FL)
Conjugate	Function (monadic)	conj	conj	Conjugate
Constant	Command	CNST	CNST	Get constant (0..78) showing temporary information (abbreviation and symbol) (CNST ___ TamNonReg menu)
Constants	MENU (ASM)	CNST	CNST	Important scientific and technical constant values (Constants preceded by "*" in programs ; Type characters 1-2 to search ; TI (temporary info) is shown in extended description)
Convergence?	Function (dyadic)	CONVG?	CONVG?	Tests convergence of X and Y using binary coded parameter for comparison mode and special numbers ; tolerance is derived from setting SDIGS (CONVG? _ TamNonReg menu)
Convert angles	MENU (43)	\angle CONV	\angle CONV	Convert angles
Convert from symmetrical components	Function (triadic)	\rightarrow abc	SYMtoA	Convert symmetrical components a0, a1, a2 in Z, Y, X to 3-phase a, b, c in Z, Y, X (\rightarrow 012)
Convert to decimal hours	Function (legacy monadic)	\rightarrow HR	\rightarrow HR	Convert to decimal hours (Assumes angle conforms to ADM)
Convert to hours, min, sec	Function (monadic)	\rightarrow h.ms	\rightarrow h.ms	Convert sexagesimal format input sequence or decimal stack value to hh:mm:ss hours (NIM input treated as sexagesimal (hh.mmss) format ; stack input treated as decimal value)
Convert to symmetrical components	Function (triadic)	\rightarrow 012	AtoSYM	Convert 3-phase a, b, c in Z, Y, X to symmetrical components a0, a1, a2 in Z, Y, X (\rightarrow abc)
Convert units	MENU	CONV	CONV	Convert units
Copy registers	Command	R-COPY	R-COPY	Copy registers (sss.nnnnn means copy registers from sss through sss + nn - 1 to registers ddd through ddd + nn - 1)
Copy Σ +NRM assignment to USER	Command	COPY \rightarrow U		Copy Σ + NORMAL assignment to USER mode assignment and activate USER mode (Does not copy Σ g assignment)
Correlation	Command	r	CORR	Correlation
Corresponds to	Character	\equiv		Character \equiv (8792)

FullName	Type	Label	Catalog	Description
Cosine	Function (monadic)	COS	COS	Cosine
Cosine	Function (monadic)	COS		Cosine
Covariance	Command	cov	COV	Covariance
Create 3x3 A-Matrix	Command	[A]	op_A	Create 3x3 A-matrix relating to Fortescue's Symmetrical Components
Cross	Function (dyadic)	x		Cross (215)
Cross (x)	Function (dyadic)	cross	CROSS	Cross product (215)
Cross markers	Command	CROSS		Use crosses as markers for the main graph
Cube	Function (monadic)	x^3	x^3	Raise to power of 3
Cube root	Function (monadic)	$\sqrt[3]{}$		Cube root (8731)
Cube root	Function (monadic)	$\sqrt[3]{x}$	$\sqrt[3]{x}$	Cube root of X
cùn to m	Function (linked ; monadic)	cùn→m		Convert cùn to meter (m)
Current number of digits	Variable (longint)	#DEC	#DEC	Current number of digits as set in DISP menu (system long integer variable, write protected)
Cursor left	Arrow	←		Move cursor left
Cursor right	Arrow	→		Move cursor right
Cursor to begin	Command	HOME←		Jump to top left of alpha input
Cursor to end	Command	END→		Jump to bottom right of alpha input
cwt to kg	Function (linked ; monadic)	cwt→kg		Convert hundredweight to kilogram (kg:)
D	Character	D		Character D (68)
d apostrophe lowercase	Character	d'		International character d' (271)
D caron	Character	Ď		International character Ď (270)
d lowercase	Character	d		Character d (100)
D stroke	Character	Đ		International character Đ (272)
d stroke lowercase	Character	đ		International character đ (273)
D47 keyboard layout	Layout (SIM)	D47		D47: Exp 2 shifts R (43S mould) /x-+ R ; The Flying Carpet ; D = double shift
Date	Command	DATE	DATE	Current date (Weekday)
Date to julian day number	Function (monadic)	D→J	D→J	Convert date to julian day number (JDN) (.0 equals noon!)
Date to stack	Function (monadic)	DATE→	DATE→	Convert date to day, month, year in stack according to DISP or CLK settings for date format
Date variables	MENU	DATES	DATES	Auto-generated catalog of variables of the specified type: date
Date-time to julian day number	Function (dyadic)	DT→J	DT→J	Convert date, time in stack to julian day number (JDN) (Date, time can be in Y, X or X, Y)
Day	Command	DAY	DAY	Day (of date)
Day month year	Setting	DMY	DMY ()	Date display mode DD.MM.YYYY (DD.MM.YYYY)
Day month year	Flag (system)	DMY		Date display mode DD.MM.YYYY (DD.MM.YYYY)
dB to fld	Function (linked ; monadic)	dB→fld		Convert decibel to field ratio (fld:)
dB to pwr	Function (linked ; monadic)	dB→pwr		Convert decibel to power ratio (pwr:)
DBLR	Function (dyadic)	DBLx	DBLx	Double word length remainder
Decimal	Setting (pgm)	DEC	DEC	Convert X to decimal and toggle decimal mode
Decimal	Function (monadic)	.d		Convert to decimal (real) value ; clear fraction mode, base mode ; convert degrees / hours / date to real ; convert complex number with zero imaginary part to real number ; convert NIM input to date (according to date format set) (decimal° : ; decimal h : ; yyyy-mm-dd :)
Decimal	Function (legacy)	→REAL	→REAL	Convert to real number
Decompose	Function (monadic)	DECOMP	DECOMP	Converts (improper) fraction to nominator in Y, denominator in X (Honours settings DENANY and DENFIX)
Decompose 3x1 matrix	Function (monadic)	M→zyx	M→zyx	Decompose 3x1 matrix to ZYX (zyx→M)
Decrement	Command	DEC	DECR	Decrement by 1 (DEC __ Tam menu)
Decrement column index	Command	J-	J-	Decrement column index
Decrement row index	Command	I-	I-	Decrement row index
Decrement skip on equal	Command	DSE	DSE	Decrement skip on equal (DSE __ Tam menu)
Decrement skip on less	Command	DSL	DSL	Decrement skip on less (DSL __ Tam menu)
Decrement skip on zero	Command	DSZ	DSZ	Decrement skip on zero (DSZ __ Tam menu)

FullName	Type	Label	Catalog	Description
Default regional formatting	Setting (pgm)	DFLT	SETDFLT	Set to default regional formats (date, time, calendar, number formatting) (First Gregorian day set: 1752-09-14)
deg to grad	Function (linked ; monadic)	deg→grad		Convert degree to gradian (untagged) (grad:)
deg to rad	Function (linked ; monadic)	deg→rad		Convert degree to radian (untagged) (rad:)
deg/s to RPM	Function (linked ; monadic)	deg/s→RPM		Convert degree per second to rotation per minute (RPM:)
Degree	Character	°		Character ° (176)
Degrees-radians-gradians	Function (cyclic ; nonpgm ; m)	DRG	DRG	Add ADM tag to untagged value in X, convert tagged value to degrees, next radians, next gradians (cyclic) (Setting tag for complex X also sets POLAR mode for X according to ADM)
Delete	Command	DELETE	EQ.DEL	Delete equation
Delete items	MENU (item)	DELITM	DELITM	Delete user defined items, selected from category (programs, variables, menus) (DELITM_ menu DELITM)
Delete items	MENU	DELITM		Delete user defined items, selected from category (programs, variables, menus) (Create user defined menus using ASSIGN)
Delete row	Command	DELR	M.DELR	Delete row from matrix
Delta	Character	Δ		Greek character Δ (916)
delta lowercase	Character	δ		Greek character δ (948)
Delta percent	Function (dyadic)	Δ%	Δ%	Delta percentage from Y to X, keeping Y on stack (Δ% :)
Delta percentage to mean	Function (monadic)	Δ% \bar{x}	Δ% \bar{x}	Delta percentage from \bar{x} to x using statistics matrix (STATS) (Δ% :)
Delta to Star (Wye)	Function (triadic)	Δ → Y	Δ → Y	Convert star connected impedances X, Y, Z to delta impedances X, Y, Z (Y → Δ)
Denominator any	Setting	DENANY	DENANY []	Any denominator will be used (fraction mode) (Result depends on setting DMX ; SBI depends on SBfrac)
Denominator any	Flag	DENANY		Set if any denominator up to D.MAX may appear (SBI depends on SBfrac)
Denominator fixed	Setting	DENFIX	DENFIX []	Only specified denominator will be used (fraction mode) (SBI depends on SBfrac)
Denominator fixed	Flag	DENFIX		Set if D.MAX is the one and only denominator allowed (SBI depends on SBfrac)
Denominator maximum	Setting (pgm)	DMX	DMX ₆₄	Fractions: D.MAX set to nn ; decimal fraction, default shown as 64ths (DMX ____ TamNonReg menu)
Determinant	Function (monadic)	DET	DET	Determinant
Determinant	Function (monadic)	M	M	Determinant
Digamma	Character	F		Greek character F (988)
digamma lowercase	Character	f		Greek character f (989)
Dimension	Function (monadic)	DIM	M.DIM	Dimension (M.DIM __ Tam menu)
Disk state	Command	DISK?	DISK?	Status of the FAT disk (Disk)
Display (program) menu	Command (PEM)	MENU	MENU	Display the programmable menu
Display (setting for) short integer	Setting	DISP_SI		Set number of lines used for the fixed X display in BASE mode (DISP_SI _ TamNonReg menu)
Display pixel	Command	PIXEL	PIXEL	Display one pixel (X, Y)
Display point	Command	POINT	POINT	Display 9 (3x3) pixels (X, Y)
Display precision	Setting (pgm)	DSP	DSP ₃	Set display mode precision (only) (DSP __ TamNonReg menu)
Display settings	MENU	DISP	DISP	Display settings
Display stack registers	Setting	DSTACK	DSTACK ₄	Set the number of stack registers to be displayed (DSTACK _ TamNonReg menu)
Distance conversion	MENU	Dist:	Dist:	Convert between units of distance
Divide	Function (dyadic)	[÷]	÷	Divide Y by X (Obelus: 247 ; Solidus: 47)
Divide into	Function (monadic)	STO/	STO/	Divide register or variable by X (STO/ __ TamStoRcl(Alpha) menu)
DM42 keyboard layout	Layout	DM42		DM42: Final compatibility layout
Dollar	Character	\$		Character \$ (36)
Dot	Character	.		Character . (8729)
Dot (•)	Function (dyadic)	dot	DOT	Dot product (8729)
Double divide	Function (triadic)	DBL/	DBL/	Double word length divide (Z - least, Y - most significant digits) / (X)
Double low-9 quotation mark	Character	„		Character „ (8222)
Double multiply	Function (dyadic)	DBL×	DBL×	Double word length multiply (result in Y - least and X - most significant digit)
Down	Symbol	↓		Move down (navigation) or arrow character (alpha selection menus) (8595)

FullName	Type	Label	Catalog	Description
Down	Command	[▼]		Scroll Down Menu (or SHOW) or Single Step
Down	Arrow	↓		Move down (navigation) or arrow character (alpha selection menus)
Draw	MENU (item)	Draw		Draw graph for equation (PLOTMENU)
Draw line	Setting	LINE		Connect graph points using line segments
Drop	Command	DROP↓	DROP	Drop one stack level (Double [⇐])
Drop Y	Command	DROPy	DROPy	Drop Y from stack
E	Character	E		Character E (69)
E (outline)	Character	<E>		Character E (8307 ; Hidden : gShiftedAIM (or Numlock) [E])
E acute	Character	É		International character É (201)
e acute lowercase	Character	é		International character é (233)
E breve	Character	Ĕ		International character Ĕ (276)
e breve lowercase	Character	ĕ		International character ĕ (277)
E caron	Character	Ě		International character Ě (282)
e caron lowercase	Character	ě		International character ě (283)
E circumflex	Character	Ê		International character Ê (202)
e circumflex lowercase	Character	ê		International character ê (234)
E diaeresis	Character	Ë		International character Ë (203)
e diaeresis lowercase	Character	ë		International character ë (235)
E dot	Character	Ė		International character Ė (278)
e dot lowercase	Character	ė		International character ė (279)
E grave	Character	È		International character È (200)
e grave lowercase	Character	è		International character è (232)
e lowercase	Character	e		Character e (101)
E macron	Character	Ē		International character Ē (274)
e macron lowercase	Character	ē		International character ē (275)
E ogonek	Character	Ę		International character Ę (280)
e ogonek lowercase	Character	ę		International character ę (281)
e to the power x	Function (monadic)	e^x	e^x	Raise e to the power in the X-register
e^{x-1}	Function (monadic)	e^{x-1}	e^{x-1}	More accurate calculation of e^{x-1} for $x \approx 0$
E47 keyboard layout	Layout (SIM)	E47		E47: Exp 2 shifts L /x++ R ; E = engineer
Earth orbit (semi major axis)	Constant (#3)	a_{\oplus}		orb.earth $a_{\oplus} = +1.495979 \times 10^{11}$ (m)
Edit equation	Command (submnu)	EDIT	EQ.EDI	Edit equation (previous equation loaded) (EIM = Equation Input Mode ; starts lowercase)
Edit matrix (named)	Command (submnu)	EDITN	M.EDIN	Edit matrix (named variable) (M.EDIN __ Tam menu)
Edit matrix (X)	MENU (item)	EDIT	M.EDI	Edit matrix (X-register) (MIM = Matrix Input Mode ; CAT.MENUS M.EDIT)
Eigenvalue	Function (monadic)	EIGVAL	EIGVAL	Eigenvalue
Eigenvector	Function (monadic)	EIGVEC	EIGVEC	Eigenvector
Electrical engineering	MENU (item)	ELEC		Electrical engineering functions and custom programs (ELEC)
Electrical engineering	MENU	ELEC	ELEC	Electrical engineering functions and custom programs
Electron magnetic moment	Constant (#66)	μ_e		mgmom.elec $\mu_e = -9.2847647043 \times 10^{-24}$ (J/T)
Electron magnetic moment / Bohr's magneton	Constant (#67)	μ_e/μ_B		r.elec.bohr $\mu_e/\mu_B = -1.00115965218128$
Electron rest mass	Constant (#23)	m_e		mass.elec $m_e = +9.1093837015 \times 10^{-31}$ (kg)
Elementary charge	Constant (#7)	e		charge.elem $e = +1.602176634 \times 10^{-19}$ (As)
Ellipsis	Character	...		Character ... (8230)
Elliptic amplitude	Function (dyadic)	$\psi(u,m)$	$\psi(u,m)$	Elliptic amplitude ($u = X$; $m = Y$)
Elliptic cosine	Function (dyadic)	$cn(u,m)$	$cn(u,m)$	Elliptic cosine ($u = X$; $m = Y$)
Elliptic delta amplitudinis	Function (dyadic)	$dn(u,m)$	$dn(u,m)$	Elliptic delta amplitudinis ($u = X$; $m = Y$)
Elliptic sine	Function (dyadic)	$sn(u,m)$	$sn(u,m)$	Elliptic sine ($u = X$; $m = Y$)
Elliptical	MENU	Ellipt	ELLIPT	Elliptical functions
Empty set	Character	∅		Character ∅ (8709)
End	Command (PEM)	END	END	End statement

FullName	Type	Label	Catalog	Description
Energy conversion	MENU	Energy:	Energy:	Convert between units of energy
Energy equivalent of m_0	Constant (#31)	m_0c^2		energy.atom $m_0c^2 = +1.4924180856 \times 10^{-10}$ (J)
Engineering display large reals	Setting (pgm)	ENGOVR	ENGOVR ()	Change display to ENG for reals too large to display in full (FF A)
Engineering notation	Setting (pgm)	ENG	ENG ()	Set numeric display mode to ENGINEERING notation with nn+1 digits (ENG __ TamNonReg menu)
Enter	Command	ENTER	ENTER↑	Enter input value to X (optionally also to Y) or push/duplicate value already in X to Y
Enter exponent	Command	[E]		Enter exponent
Entry RPN	Setting	eRPN	eRPN [.]	Entry RPN
Entry RPN	Command	eRPN?	eRPN?	Entry RPN mode : 1 ; classic RPN mode : 0 ; set by ON (For programming purposes)
Entry RPN off	Setting (pgm)	eRPNoff	RPN	Set stack to classic RPN mode (For programming purposes)
Entry RPN on	Setting (pgm)	eRPNon	eRPN	Set stack to entry RPN mode (no DUP) (For programming purposes)
Epsilon	Character	Ε		Greek character Ε (917)
epsilon lowercase	Character	ε		Greek character ε (949)
epsilon tonos lowercase	Character	έ		Greek character έ (941)
Equals	Symbol	=		Equals (61)
Equation	MENU	EQN	EQN	Equation editor (Equation entry default lowercase)
Equation editor	MENU (item)	NEW	EQ.NEW	Create new equation (previous equation pushed) (EIM = Equation Input Mode ; starts lowercase)
Equation editor	MENU	NEW		Create new equation (previous equation pushed) (EIM = Equation Input Mode ; starts lowercase)
Error	Command (PEM)	ERR	ERR	Raise error and show error message (ERR __ TamNonReg menu)
Error function	Function (monadic)	erf	erf	Error function
Estimates	Character	≐		Character ≐ (8793)
Eta	Character	Η		Greek character Η (919)
eta lowercase	Character	η		Greek character η (951)
eta tonos lowercase	Character	ή		Greek character ή (942)
Euclidean norm	Function (monadic)	ENORM	ENORM	Euclidean norm
Euler-Mascheroni constant	Constant (#57)	γ_{EM}		c.eul.masc $\gamma_{EM} = +5,772156649015328606065120900824024 \times 10^{-1}$
Euler's Beta function	Function (dyadic)	$\beta(x,y)$	$\beta(x,y)$	Euler's Beta function
Euler's e	Constant (#8)	e		e.euler $e = +2,718281828459045235360287471352662$
Euler's formula	Function (monadic)	e^{ix}	e^{ix}	Rotate complex unit vector by X radians : $e^{ix} = \cos(x) + i \sin(x)$ (i or j displayed in stack and on soft buttons)
Euro	Character	€		Character € (8364)
European formatting	Setting (pgm)	EUROPE	SETEUR	Set to European regional formats (date, time, calendar, number formatting) (First Gregorian day set: 15.10.1582)
Even?	Function (monadic)	EVEN?	EVEN?	Test X is integer AND even
Exchange	Character	↔		Character ↔ (8644)
Exchange real and imaginary part	Command	Re↔Im	Re↔Im	Exchange real and imaginary part
Exclusive NOR	Function (dyadic)	XNOR	XNOR	Logical exclusive NOR (bitwise)
Execute	Command	XEQ	XEQ	Execute function or program (XEQ __ TamLbl(Alpha) menu)
Execute XEQM command	Command	X.XEQ	X.XEQ	Execute XEQM command in X-register
Exit	Command	EXIT		EXIT
Exit all	Command	EXITall	EXITALL	Exit all menus ; return to MyMenu (interactive mode) ; exit VarMNU (program mode)
Expanded inverse tangent	Function (dyadic)	ATAN2	ATAN2	Arc tangent of Y / X
Exponent	Symbol	^		Raise to power (94)
Exponent	Function (monadic)	EXPT	EXPT	Exponent of number in the X-register
Exponential	MENU	EXP	EXP	Exponential functions
Exponential (inverse)	Function (tbd)	Expon^{-1}	Expon^{-1}	Exponential probability inverse function
Exponential cdf (lower)	Function (tbd)	Expon_\downarrow	Expon_\downarrow	Exponential cumulative distribution (lower tail)
Exponential cdf (upper)	Function (tbd)	Expon_\uparrow	Expon_\uparrow	Exponential cumulative distribution (upper tail)
Exponential distribution	MENU	Expon:	Expon:	Exponential probability distribution (RegI = λ = rate parameter)
Exponential fit	Setting	ExpF	ExpF []	Exponential curve fitting
Exponential pdf	Function (tbd)	Expon_p	Expon_p	Exponential probability density function

FullName	Type	Label	Catalog	Description
Export program	Command	XPORTP	XPORTP	Export program to text file in FAT (XPORTP _ TamLabel ; DMCP : File save dialog (PROGRAMS))
Extended fractions	Setting	EXFRAC	EXFRAC []	Extended fractions, helper mode to find approximate multiples or fractions of the irrational numbers $\sqrt{2}$, $\sqrt{3}$, $\sqrt{5}$, ϕ , π and e (\approx <approximation> ; SBI depends on SBfrac)
Extended functions	MENU	X.FN	X.FN	Extended functions (Bessel, Bernoulli, Gamma, Elliptical, Orthogonal, etc.)
F	Character	F		Character F (70)
f lowercase	Character	f		Character f (102)
f'(x)	Function (monadic)	f'(x)	f'(x)	First derivative of f at x (f'(x) __ TamLbl(Alpha) menu)
f''(x)	Function (monadic)	f''(x)	f''(x)	Second derivative of f at x (f''(x) __ TamLbl(Alpha) menu)
Factorial ; $\Gamma(x+1)$	Symbol	!		For integers : x! ; for reals : $\Gamma(x+1)$ (33)
Factorial ; $\Gamma(x+1)$	Function (monadic)	x!	x!	For integers : x! ; for reals : $\Gamma(x+1)$ (Max integer: 450 ; max real : 2123.549 956 662 463 236 31 ; integers > max are converted to reals)
Faraday constant	Constant (#9)	F		c.faraday F = +9.64853321233100184 $\times 10^4$ (As/mol)
fathom to m	Function (linked ; monadic)	fathom→m		Convert fathom to meter (m:)
Feigenbaum alpha	Constant (#10)	F _α		α.feigenbm F _α = +2,502907875095892822283902873218216
Feigenbaum delta	Constant (#11)	F _δ		δ.feigenbm F _δ = +4,669201609102990671853203820466202
femto	Command (nonpgm)	•f		Factor 10 ⁻¹⁵
fēn to m	Function (linked ; monadic)	fēn→m		Convert fēn to meter (m:)
Fibonacci	Function (monadic)	FIB	FIB	Fibonacci number n, where n = X
Fill stack	Command	FILL	FILL	Fill stack with value in the X-register
Financial	MENU	FIN	FIN	Financial calculations including time value of money (TVM)
Fine-structure constant	Constant (#55)	α		c.finestruc α = +7,2973525693 $\times 10^{-3}$
fir to kg	Function (linked ; monadic)	fir→kg		Convert firkin to kilogram (kg:)
fir to lb.	Function (linked ; monadic)	fir→lb.		Convert firkin to pound (lb.:)
First derivative	MENU	f'	f'	First derivative
Fisher's F (inverse)	Function (tbd)	F ⁻¹ (p)	F ⁻¹ (p)	Fisher's F probability inverse function
Fisher's F cdf (lower)	Function (monadic)	F _▲ (x)	F _▲ (x)	Fisher's F cumulative distribution (lower tail)
Fisher's F cdf (upper)	Function (monadic)	F _△ (x)	F _△ (x)	Fisher's F cumulative distribution (upper tail)
Fisher's F distribution	MENU	F:	F:	Fisher's F probability distribution (RegI = d ₁ = degree of freedom ; RegJ = d ₂ = degree of freedom)
Fisher's F pdf	Function (monadic)	F _p (x)	F _p (x)	Fisher's F probability density function
FIX-SCI-ENG	Setting (cyclic)	FSE		Display mode cycling (Fixed, Scientific, Engineering)
Fixed notation	Setting (pgm)	FIX	FIX ()	Set numeric display mode to FIXed notation with nn+1 digits (FIX __ TamNonReg menu)
Flag browser	Browser	FLGS	FLGS	Show all flags on one page (0 = clear, 1 = set) ; show status page(s) on Up/Dn (Compare FLAGS.STATUS)
Flag clear?	Command	FC?	FC?	Test flag clear? (FC? __ TamFlag menu)
Flag clear? and clear	Command	FC?C	FC?C	Test flag clear? and clear (FC?C __ TamFlag menu)
Flag clear? and flip	Command	FC?F	FC?F	Test flag clear? and flip (FC?F __ TamFlag menu)
Flag clear? and set	Command	FC?S	FC?S	Test flag clear? and set (FC?S __ TamFlag menu)
Flag set?	Command	FS?	FS?	Test flag set? (FS? __ TamFlag menu)
Flag set? and clear	Command	FS?C	FS?C	Test flag set? and clear (FS?C __ TamFlag menu)
Flag set? and flip	Command	FS?F	FS?F	Test flag set? and flip (FS?F __ TamFlag menu)
Flag set? and set	Command	FS?S	FS?S	Test flag set? and set (FS?S __ TamFlag menu)
Flags	MENU	FLAG	FLAG	Setting, clearing and testing flags
Flattening factor	Constant (#49)	Sf ⁻¹		f.flatteng Sf ⁻¹ = +2.98257223563 $\times 10^2$
fld to dB	Function (linked ; monadic)	fld→dB		Convert field ratio to decibel (dB:)
Flip bit	Function (monadic)	FB	FB	Flip bit n (FB __ TamNonReg menu)
Flip case (one character)	Command	(FlipChar)		Flip case (one character) (fShiftedAIM [f g])
Flip flag	Command	FF	FF	Flip flag (toggle) (FF __ TamFlag menu)
Floor	Function (monadic)	floor	floor	Floor (type real)
floz _{UK} to ml	Function (linked ; monadic)	floz _{UK} →ml		Convert UK fluid ounce to milliliter (ml:)

FullName	Type	Label	Catalog	Description
floz _{US} to ml	Function (linked ; monadic)	floz _{US} →ml		Convert US fluid ounce to milliliter (ml:)
Font browser	Browser	FBR	FBR	Browse system fonts (character tables)
Force & Pressure conversion	MENU	F&p:	F&p:	Convert between units of force and pressure
Force autoscale	Setting	X:Y=1		PLSTAT only: force the autoscales on x- and y-axis to be the same
FP group size	Setting (pgm)	FPGRP	FPGRP ₃	Set fractional part group size (2..9) (FPGRP _ TamNonReg menu)
FP separator comma	Setting (pgm)	COM,	FCOM, ()	Set fractional part separator to comma
FP separator dot	Setting (pgm)	DOT.	FDOT. ()	Set fractional part separator to dot
FP separator double space	Setting (pgm)	WSPC _{..}	FWSPC _{..} ()	Set fractional part separator to double space
FP separator narrow space	Setting (pgm)	NSPC _.	FNSPC _. ()	Set fractional part separator to narrow space
FP separator narrow tick	Setting (pgm)	TICK'	FTICK' ()	Set fractional part separator to narrow tick
FP separator period	Setting (pgm)	PER.	FPER. ()	Set fractional part separator to period
FP separator space	Setting (pgm)	SPC _.	FSPC _. (•)	Set fractional part separator to space
FP separator tick	Setting (pgm)	WTICK'	FWTICK' ()	Set fractional part separator to tick
FP separator underscore	Setting (pgm)	UNDR_	FUNDR_ ()	Set fractional part separator to underscore
FP separator wide comma	Setting (pgm)	WCOM,	FWCOM, ()	Set fractional part separator to wide comma
FP separator wide dot	Setting (pgm)	WDOT•	FWDOT• ()	Set fractional part separator to wide dot
FP separator wide period	Setting (pgm)	WPER.	FWPER. ()	Set fractional part separator to wide period
Fraction	Flag	FRACT		Fraction mode (proper, improper fractions, tolerance determined by DMX) ; SBI •¶ indicates that approximate multiples or fractions are used, of the irrational numbers $\sqrt{2}$, $\sqrt{3}$, $\sqrt{5}$, ϕ , π and e (with tolerance 10^{-24}) (Prefix < or > (or \approx when •¶ is lit) ; /n or /max denotes maximum denominator (set by DMX) ; SBI depends on SBfrac)
Fraction (mode)	Setting (cyclic ; stack)	a ^b /c		Set and cycle fraction mode (proper, improper fractions, tolerance determined by DMX) ; SBI •¶ indicates that approximate multiples or fractions are used, of the irrational numbers $\sqrt{2}$, $\sqrt{3}$, $\sqrt{5}$, ϕ , π and e (with tolerance 10^{-24}) ; reset by [.d] (gShifted [LOG]) (Double [.] ; Info : Prefix < or > (or \approx when •¶ is lit) ; /n or /max denotes maximum denominator (set by DMX) ; SBI depends on SBfrac)
Fraction show register name	Setting	FRCSRN	FRCSRN []	Fractions are shown with register names (x, y, z, t < or = or > or \approx)
Fraction show register name	Flag	FRCSRN		Fractions are shown with register names (x, y, z, t < or = or > or \approx)
Fractional part	Function (monadic)	FP	FP	Fractional part (#F (closed number))
Fractional part?	Function (monadic)	FP?	FP?	Test X has nonzero fractional part
ft. to m	Function (linked ; monadic)	ft.→m		Convert foot to meter (m:)
ft/s to km/h	Function (linked ; monadic)	ft/s→km/h		Convert foot per second to kilometer per hour (km/h:)
ft/s to m/s	Function (linked ; monadic)	ft/s→m/s		Convert foot per second to meter per second (m/s:)
ftn to s	Function (linked ; monadic)	ftn→s		Convert fortnight to second (s:)
Function keys full cycle	Setting	F.1234		Longpress control : full Function key longpress cycle
Function keys g-shortcut	Setting	g.2Tp	g.2Tp [•]	Allow double tapping the FN-keys for a g-function (Blocked for (navigation) arrows in editors)
Function keys skip fg	Setting	F.14		Longpress control : skip f and g Function key longpress cycle
Function keys skip g	Setting	F.124		Longpress control : skip g Function key longpress cycle
Functions	MENU (ASM)	FCNS	FCNS	Catalog of all calculator functions (Type characters 1-2 to search)
fur to m	Function (linked ; monadic)	fur→m		Convert furlong to meter (m:)
fur/ftn to km/h	Function (linked ; monadic)	fur/ftn→km/h		Convert furlong per fortnight to kilometer per hour (km/h:)
fur/ftn to m/s	Function (linked ; monadic)	fur/ftn→m/s		Convert furlong per fortnight to meter per second (m/s:)
fur/ftn to mph	Function (linked ; monadic)	fur/ftn→mph		Convert furlong per fortnight to mile per hour (mph:)
Future value	Variable (real)	FV	FV	Future value (reserved real variable)
G	Character	G		Character G (71)
G breve	Character	Ĝ		International character Ĝ (286)
g breve lowercase	Character	ĝ		International character ĝ (287)
g lowercase	Character	g		Character g (103)
g to carat	Function (linked ; monadic)	g→carat		Convert gram to carat (carat:)

FullName	Type	Label	Catalog	Description
g to oz	Function (linked ; monadic)	g→oz		Convert gram to ounce (oz:)
g to tr.oz	Function (linked ; monadic)	g→tr.oz		Convert gram to troy ounce (tr.oz:)
g _d	Function (monadic)	g _d	g _d	Gudermannian function
g _d ⁻¹	Function (monadic)	g _d ⁻¹	g _d ⁻¹	Inverse Gudermannian function
gal _{UK} to l	Function (linked ; monadic)	gal _{UK} →l		Convert UK gallon to liter (l:)
gal _{US} to l	Function (linked ; monadic)	gal _{US} →l		Convert US gallon to liter (l:)
Gamma	Character	Γ		Greek character Γ (915)
gamma lowercase	Character	γ		Greek character γ (947)
Gauss fit	Setting	GaussF	GaussF []	Gauss curve fitting
Geocentric gravitational constant	Constant (#16)	GM _⊕		c.grav.geo GM _⊕ = +3.986004418 × 10 ¹⁴ (m ³ /s ²)
Geometric (inverse)	Function (tbd)	Geom ⁻¹	Geom ⁻¹	Geometric probability inverse function
Geometric cdf (lower)	Function (tbd)	Geom _p	Geom _p	Geometric cumulative distribution (lower tail)
Geometric cdf (upper)	Function (tbd)	Geom _p	Geom _p	Geometric cumulative distribution (upper tail)
Geometric distribution	MENU	Geom:	Geom:	Geometric probability distribution (RegI = p = probability)
Geometric means	Command	\bar{x}_G	\bar{x}_G	Geometric means (\bar{x}_G ; \bar{y}_G = (2 stack levels))
Geometric pdf	Function (tbd)	Geom _p	Geom _p	Geometric probability density function
Get annual interest rate	Command (nonpgm)	R I%/a		Get annual interest rate
Get annual periods	Command (nonpgm)	R PER/a		Get annual number of payment or compounding periods for loan or investment
Get future value	Command (nonpgm)	R FV		Get future value
Get hide (setting)	Command	HIDE?	HIDE?	Current setting of HIDE
Get Julian-Gregorian (transition)	Command	J/G?	J/G?	Get the day that Julian date changes over to Gregorian date (Set by J/G)
Get payment	Command (nonpgm)	R PMT		Get payment
Get present value	Command (nonpgm)	R PV		Get present value
Get range (setting)	Command	RANGE?	RANGE?	Maximum number exponent (range) ; set by RNG
Get submatrix	Command	GETM	M.GET	Get submatrix
Get total periods	Command (nonpgm)	R N _{PER}		Get total number of payment or compounding periods for loan or investment
Giga	Command (nonpgm)	*G		Factor 10 ⁹
Go to	Command	GTO	GTO	Go to local/global label or line (GTO __ TamLbl(Alpha) menu)
Go to label or step	Command	GTO.	GTO.	Go to label or step (GTO. ____ GTO menu)
Go to matrix column	Command	GOTO Column		Go to matrix column for GOTO (GOTO Column ____ TamNonReg menu)
Go to matrix element	Command	GOTO	M.GOTO	Go to matrix element using GOTO Row ; GOTO Column (GOTO Row ; GOTO Column)
Go to matrix row	Command	GOTO Row		Go to matrix row for GOTO (GOTO Row ____ TamNonReg menu)
Golden ratio	Constant (#73)	φ		r.golden φ = +1,618033988749894848204586834365638 (φ = (1 + √5) / 2)
grad to deg	Function (linked ; monadic)	grad→deg		Convert gradian to degree (untagged) (deg:)
grad to rad	Function (linked ; monadic)	grad→rad		Convert gradian to radian (untagged) (rad:)
Graphics mode	Variable (longint)	GRAMOD	GRAMOD	Graphics display mode for AGRAPH (reserved long integer variable) (OR: 0 ; SET: 1 ; OFF: 2 ; XOR: 3)
Graphing	MENU	GRAPH		Graphing functions
Greater or equal than	Character	≥		Character ≥ (8805)
Greater than	Character	>		Character > (62)
Greatest common divisor	Function (dyadic)	GCD	GCD	Greatest common divisor of X and Y
Gregorian year	Constant (#0)	a		yr.gregor a = +3.652425 × 10 ² (d)
Grow (matrix edit)	Setting	GROW	M.GROW ()	Matrix edit in grow mode (SBI depends on SBmx)
Grow (matrix edit)	Flag	GROW		Matrix edit in growing mode (SBI depends on SBmx)
H	Character	H		Character H (72)
h lowercase	Character	h		Character h (104)
h stroke lowercase	Character	ħ		International character ħ (295)
ha to acre	Function (linked ; monadic)	ha→acre		Convert hectare to acre (acre:)
ha to acre _{US}	Function (linked ; monadic)	ha→acre _{US}		Convert hectare to US acre (acre _{US} :)

FullName	Type	Label	Catalog	Description
ha to m ²	Function (linked ; monadic)	ha→m ²		Convert hectare to square meter (m ² :)
Harmonic means	Command	\bar{x}_H	\bar{x}_H	Harmonic means (\bar{x}_H ; $\bar{y}_H = (2 \text{ stack levels})$)
Hash	Character	#		Character # (35)
Hermite polynomials (physics)	Function (dyadic)	H _{nP}	H _{nP}	Hermite polynomials (physics) (x = X ; n = Y)
Hermite polynomials (probability)	Function (dyadic)	H _n	H _n	Hermite polynomials (probability) (x = X ; n = Y)
Hexadecimal	Setting (pgm)	HEX	HEX	Convert X to hexadecimal and toggle hexadecimal mode
Hexadecimal base	Shortcut (TAM)	#16		TAM shortcut for setting hexadecimal base (Shortcut H(ex) (C47.32.41))
Hide small values	Setting (pgm)	HIDE	HIDE ₀	Hide (i.e. display '0.' instead of) all real numbers or parts with absolute values < 10 - n with n = IP(x), 12 ≤ n ≤ 99 ; useful e.g. in matrices ; reset by HIDE 0 (HIDE __ TamNonReg menu)
High bin	Command	↑BIN	↑BIN	High bin (nBINS ; ; ↓BIN ; ; ↑BIN : (3 stack levels))
Histogram	MENU (item)	H PLOT	H PLOT	Histogram plotting (H PLOT)
Histogram	MENU	HIST		Histogram functions
Histogram	MENU	H PLOT	H PLOT	Histogram plotting
Histogram matrix	Variable (matrix)	HISTO	HISTO	Reserved matrix variable Histogram (HISTO) (Create from STATS using menu HIST)
Histogram Normal	Command	HNORM	HNORM	Fit Gauss distribution through HISTO data
Histogram X	Command	HISTOX	HISTOX	Evaluate first column of STATS and store in HISTO (nBINS ; ; ↓BIN ; ; ↑BIN : (3 stack levels))
Histogram Y	Command	HISTOY	HISTOY	Evaluate second column of STATS and store in HISTO (nBINS ; ; ↓BIN ; ; ↑BIN : (3 stack levels))
HOME	MENU	HOME	HOME	HOME menu (Triple [f/G]) (HOME.3 ON))
Home menu fff shortcut	Setting	HOME.3	HOME.3 [.]	HOME menu activated by triple shift
HOME menu shown	Setting	HOME	_HOME []	HOME menu shown (when all menus are exited using EXIT)
Hourglass	Character	⌚		Character ⌚ (8987)
Hours	Function (monadic)	HOUR	HOUR	Hours (of time)
HP style base	Setting	BASE _{HP}	BASE _{HP} [.]	Set for the classic (HP) convention that all stack registers are changed at once when base mode is changed (shortint values only)
HP style base	Flag	BASE _{HP}		Set for the classic (HP) convention that all stack registers are changed at once when base mode is changed (shortint values only)
hp _E to W	Function (linked ; monadic)	hp _E →W		Convert electrical horsepower to Watt (W:)
hp _M to W	Function (linked ; monadic)	hp _M →W		Convert metric horsepower to Watt (W:)
hp _{UK} to W	Function (linked ; monadic)	hp _{UK} →W		Convert UK horsepower to Watt (W:)
Humorous conversions	MENU	FFF+	FFF+	Conversions to and from the furlong-firkin-fortnight (FFF) system (and beardseconds)
Hyperbolic cosine	Function (monadic)	cosh	cosh	Hyperbolic cosine
Hyperbolic fit	Setting	HypF	HypF []	Hyperbolic curve fitting
Hyperbolic sine	Function (monadic)	sinh	sinh	Hyperbolic sine
Hyperbolic tangent	Function (monadic)	tanh	tanh	Hyperbolic tangent
Hyperfine transition frequency of ¹³³ Cs	Constant (#59)	Δν _{Cs}		frq.hypf.cs Δν _{Cs} = +9.19263177 × 10 ⁹ (Hz)
Hypergeometric (inverse)	Function (tbd)	Hyper ⁻¹	Hyper ⁻¹	Hypergeometric probability inverse function
Hypergeometric cdf (lower)	Function (tbd)	Hyper _▲	Hyper _▲	Hypergeometric cumulative distribution (lower tail)
Hypergeometric cdf (upper)	Function (tbd)	Hyper _▲	Hyper _▲	Hypergeometric cumulative distribution (upper tail)
Hypergeometric distribution	MENU	Hyper:	Hyper:	Hypergeometric probability distribution (RegI = N = population size ; RegJ = n = sample size ; RegK = K = number of special items in the pool)
Hypergeometric pdf	Function (tbd)	Hyper _p	Hyper _p	Hypergeometric probability density function
I	Character	I		Character I (73)
I acute	Character	İ		International character İ (205)
i acute lowercase	Character	ı		International character ı (237)
I breve	Character	İ̇		International character İ̇ (300)
i breve lowercase	Character	ı̇		International character ı̇ (301)
I circumflex	Character	Î		International character Î (206)

FullName	Type	Label	Catalog	Description
i circumflex lowercase	Character	î		International character î (238)
I diaeresis	Character	Ï		International character Ï (207)
i diaeresis lowercase	Character	ï		International character ï (239)
I dot	Character	İ		International character İ (304)
i dotless lowercase	Character	ı		International character ı (305)
I grave	Character	Ï		International character Ï (204)
i grave lowercase	Character	ï		International character ï (236)
i lowercase	Character	i		Character i (105)
I macron	Character	Ī		International character Ī (298)
i macron lowercase	Character	ī		International character ī (299)
I ogonek	Character	Į		International character Į (302)
i ogonek lowercase	Character	į		International character į (303)
I _{xyz}	Function (triadic)	I _{xyz}	I _{xyz}	Regularised (incomplete) Beta function
Ignore One Error	Flag	IGN1ER		Set for calculator ignoring just 1 arbitrary error ; subsequently clears IGN1ER
Imaginary i	Setting (pgm)	CPX <i>i</i>	CPX <i>i</i> (•)	Set for the letter i representing the imaginary number (<i>i</i> or <i>j</i> displayed in stack and on soft buttons)
Imaginary j	Setting (pgm)	CPX <i>j</i>	CPX <i>j</i> ()	Set for the letter j representing the imaginary number (<i>i</i> or <i>j</i> displayed in stack and on soft buttons)
Imaginary number	Command	<i>i</i>	op_ <i>i</i>	Complex number <i>i</i> ; displayed according to flag CPX <i>j</i> (default: <i>i</i>) (In NIM, works like CC ; RECT input assumed always)
Imaginary number	Command	<i>i</i>		Complex number <i>i</i> ; displayed according to flag CPX <i>j</i> (default: <i>i</i>)
Imaginary part	Function (monadic)	Im	Im	Imaginary part of complex number
in. to brds	Function (linked ; monadic)	in.→brds		Convert inch to beardsecond (brds:)
in. to mm	Function (linked ; monadic)	in.→mm		Convert inch to millimeter (mm:)
in.Hg to Pa	Function (linked ; monadic)	in.Hg→Pa		Convert inch of Mercury to Pascal (Pa:)
Incomplete elliptic integral (1st)	Function (dyadic)	F(φ,m)	F(φ,m)	Incomplete elliptic integral of the 1st kind (φ = X ; m = Y)
Incomplete elliptic integral (2nd)	Function (dyadic)	E(φ,m)	E(φ,m)	Incomplete elliptic integral of the 2nd kind (φ = X ; m = Y)
Increment	Command	INC	INC	Increment by 1 (INC __ Tam menu)
Increment column index	Command	J+	J+	Increment column index
Increment row index	Command	I+	I+	Increment row index
Increment skip on equal	Command	ISE	ISE	Increment skip on equal (ISE __ Tam menu)
Increment skip on greater	Command	ISG	ISG	Increment skip on greater (ISG __ Tam menu)
Increment skip on zero	Command	ISZ	ISZ	Increment skip on zero (ISZ __ Tam menu)
Index the matrix	Command	INDEX	INDEX	Index the matrix (INDEX __ Tam menu)
Indian formatting	Setting (pgm)	INDIA	SETIND	Set to Indian regional formats (date, time, calendar, number formatting) (First Gregorian day set: 14.09.1752)
Indirection	Command (TAM)	→		Presented in TAM menus for commands accessing indirect input
Infinities	Command	±∞?	±∞?	Show whether X-register contains positive or negative infinite value
Infinity	Character	∞		Character ∞ (8734)
Information	MENU	INFO	INFO	System information and some information about the value in the X-register
Input	Command (PEM)	INPUT	INPUT	Halt program execution, push current value and accept input for variable or register (INPUT __ Tam menu)
Input complex	Setting	i CPX		New number input is put on the stack as complex
Input longint	Setting	i LI		New numbers are put on the stack as long integer
Input longint, real	Setting	i LI/RL		Input of Long Integer and Reals ; standard automatic determination of which, using the decimal radix as differentiator to select Real otherwise long integer
Input real	Setting	i REAL		New number input is put on the stack as real
Input/Output	MENU	I/O	I/O	Input/output functions
Insert row	Command	INSR	M.INSR	Insert row into matrix
Integer divide	Function (dyadic)	IDIV	IDIV	Integer divide

FullName	Type	Label	Catalog	Description
Integer divide and remainder	Function (dyadic)	IDIVR	IDIVR	Integer divide (X) and remainder (Y)
Integer part	Function (monadic)	IP	IP	Integer part (type real) (#1 (closed number))
Integer product (programmable)	Command	iΠ _n	iΠ _n	Integer product using specified program, with iteration counter, interrupt by keypress (iΠ _n _ TamLbl(Alpha) menu ; <from> ENTER <to> ENTER <step>)
Integer sign mode	Command	ISM?	ISM?	Sign mode for short integers ; set by UNSIGN ; SIGNMT
Integer sign mode	Variable (longint)	ISM	ISM	Integer sign mode (reserved long integer variable) (SIGNMT: s ; UNSIGN: u ; 1COMPL: 1 ; 2COMPL: 2)
Integer sum (programmable)	Command	iΣ _n	iΣ _n	Integer sum using specified program, with iteration counter, interrupt by keypress (iΣ _n _ TamLbl(Alpha) menu ; <from> ENTER <to> ENTER <step>)
Integer?	Function (monadic)	INT?	INT?	Test X has zero fractional part
Integers	MENU	INTS	INTS	Short integer functions
Integral	Command	∫	∫	Calculates integral (∫≈)
Integral	Character	∫		Character ∫ (8747)
Integral f	MENU	∫ f	∫ f	Integral f
Integral f dx	MENU	∫ f dx	∫ f dx	Integral f dx (∫ f dx _ TamLbl(Alpha) menu)
Integrator Running	Flag (system)	INTING		Integrator is running
Interquartile range	Command	x _{IQR}	x _{IQR}	Interquartile range for both X and Y ; this is equal to Q.3 - Q.1 (iqr _x ; iqr _y = (2 stack levels))
Inverse	Character	-1		Character -1 (8306)
Inverse hyperbolic cosine	Function (monadic)	arcosh	arcosh	Inverse hyperbolic cosine
Inverse hyperbolic sine	Function (monadic)	arsinh	arsinh	Inverse hyperbolic sine
Inverse hyperbolic tangent	Function (monadic)	artanh	artanh	Inverse hyperbolic tangent
Inverse of .ms	Function (monadic)	.ms ⁻¹	.ms ⁻¹	Convert hh:mm:ss hours or dd°mm' ss" degrees to sexagesimal format number (untag) (hh/ddd.mmss:)
Invert matrix	Function (monadic)	[M] ⁻¹	[M] ⁻¹	Inverse of matrix
Inverted exclamation mark	Character	¡		Character ¡ (161)
Inverted question mark	Character	¿		Character ¿ (191)
Iota	Character	ι		Greek character ι (921)
Iota dialytika	Character	ϊ		Greek character ι̂ (938)
Iota dialytika lowercase	Character	ι̂		Greek character ι̂ (970)
Iota dialytika tonos lowercase	Character	ι̂̂		Greek character ι̂̂ (912)
Iota lowercase	Character	ι		Greek character ι (953)
Iota tonos lowercase	Character	ι̂		Greek character ι̂ (943)
IP first group extension	Setting (pgm)	IPGRP1x	IPGRP1x ₀	Extend first group to allow one additional digit up to maximum specified by parameter (IPGRP1x _ TamNonReg menu)
IP first group size	Setting (pgm)	IPGRP1	IPGRP1 ₀	Set integer part first group size ; parameter 0 means follow IPGRP (IPGRP1 _ TamNonReg menu)
IP group size	Setting (pgm)	IPGRP	IPGRP ₃	Set integer part group size (2..9) ; not for SCI or ENG notation (IPGRP _ TamNonReg menu)
IP separator comma	Setting (pgm)	COM,	ICOM, ()	Set integer part separator to comma
IP separator dot	Setting (pgm)	DOT.	IDOT. ()	Set integer part separator to dot
IP separator double space	Setting (pgm)	WSPC..	IWSPC.. ()	Set integer part separator to double space
IP separator narrow space	Setting (pgm)	NSPC.	INSPC. ()	Set integer part separator to narrow space
IP separator narrow tick	Setting (pgm)	TICK'	ITICK' ()	Set integer part separator to narrow tick
IP separator period	Setting (pgm)	PER.	IPER. ()	Set integer part separator to period
IP separator space	Setting (pgm)	SPC.	ISPC. ()	Set integer part separator to space
IP separator tick	Setting (pgm)	WTICK'	IWTICK' ()	Set integer part separator to tick
IP separator underscore	Setting (pgm)	UNDR_	IUNDR_ ()	Set integer part separator to underscore
IP separator wide comma	Setting (pgm)	WCOM,	IWCOM, ()	Set integer part separator to wide comma
IP separator wide dot	Setting (pgm)	WDOT.	IWDOT. ()	Set integer part separator to wide dot
IP separator wide period	Setting (pgm)	WPER.	IWPER. ()	Set integer part separator to wide period

FullName	Type	Label	Catalog	Description
Γ_p	Function (dyadic)	Γ_p	Γ_p	Regularised Gamma function (P)
Γ_q	Function (dyadic)	Γ_q	Γ_q	Regularised Gamma function (Q)
J	Character	J		Character J (74)
j lowercase	Character	j		Character j (106)
J to Btu	Function (linked ; monadic)	J→Btu		Convert Joule to British Thermal Unit (Btu:)
J to cal	Function (linked ; monadic)	J→cal		Convert Joule to calorie (cal:)
J to Wh	Function (linked ; monadic)	J→Wh		Convert Joule to Watt-hour (Wh:)
$J_y(x)$	Function (monadic)	$J_y(x)$	$J_y(x)$	Bessel function of the 1st kind and order y
Jacobi's Zeta	Function (dyadic)	Z(ϕ ,m)	Z(ϕ ,m)	Jacobi's Zeta ($\phi = X$; $m = Y$)
Japanese formatting	Setting (pgm)	JAPAN	SETJPN	Set to Japanese regional formats (date, time, calendar, number formatting) (First Gregorian day set: 1873-01-01)
jīn to kg	Function (linked ; monadic)	jīn→kg		Convert jīn to kilogram (kg:)
Josephson constant	Constant (#21)	K_J		c.josephsn $K_J = +4,835978484169836324476582850545281 \times 10^{14}$ (Hz/V)
Julian day number to date-time	Command	J→DT	J→DT	Convert julian day number (JDN) to date, time in stack
Julian-Gregorian transition 1582	Setting (pgm)	JG.1582	JG.1582	Set Julian-Gregorian transition date to 1582-10-15 (First Gregorian day set: 1582-10-15)
Julian-Gregorian transition 1752	Setting (pgm)	JG.1752	JG.1752	Set Julian-Gregorian transition date to 1752-09-14 (First Gregorian day set: 1752-09-14)
Julian-Gregorian transition 1873	Setting (pgm)	JG.1873	JG.1873	Set Julian-Gregorian transition date to 1873-01-01 (First Gregorian day set: 1873-01-01)
Julian-Gregorian transition 1949	Setting (pgm)	JG.1949	JG.1949	Set Julian-Gregorian transition date to 1949-10-01 (First Gregorian day set: 1949-10-01)
K	Character	K		Character K (75)
k lowercase	Character	k		Character k (107)
Kappa	Character	K		Greek character K (922)
kappa lowercase	Character	κ		Greek character κ (954)
Key	Command (PEM)	KEY	KEY	Used in programs to form program step for KEYG → KEY nn GTO __ or KEYX → KEY nn XEQ __ (KEY __ TamNonReg menu)
Key execute	Command (PEM)	KEYX	KEYX	Used in programs to specify label to execute when soft button (1-18), [▲] (19), [▼] (20) or EXIT (21) is pressed (KEY __ TamKey menu ; KEY nn XEQ __ TamLbl(Alpha) menu)
Key go to	Command (PEM)	KEYG	KEYG	Used in programs to specify label to go to when soft button (1-18), [▲] (19), [▼] (20) or EXIT (21) is pressed (KEY __ TamKey menu ; KEY nn GTO __ TamLbl(Alpha) menu)
Key pressed?	Command	KEY?	KEY?	Test key was pressed (store keycode in register) (KEY? __ Tam menu)
Key type	Function (monadic)	KTYP?	KTYP?	Key type for key code returned by KEY? (KTYP? __ Tam menu)
Keyboard (USER) reset	Command	K.RESET		Reset keyboard user key assignments (C47 All USER keys cleaned)
Keyboard map	Browser	KEYMAP		Show keyboard map for USER ; press and hold key [.] to view STD mode (also showing special assignment from menu Σ+NRM) (Browse key shifts using [▲] and [▼])
Keys	MENU	KEYS	KEYS	Keyboard layouts (Selection of a non-default layout sets USER mode ; Selection of the default layout (C47) clears USER mode ; Switching layouts cleans all user assignments! (use SAVE to backup))
kg to cwt	Function (linked ; monadic)	kg→cwt		Convert kilogram to hundredweight (cwt:)
kg to fir	Function (linked ; monadic)	kg→fir		Convert kilogram to firkin (fir:)
kg to jīn	Function (linked ; monadic)	kg→jīn		Convert kilogram to jīn (jīn:)
kg to lb.	Function (linked ; monadic)	kg→lb.		Convert kilogram to pound (lb.:)
kg to liǎng	Function (linked ; monadic)	kg→liǎng		Convert kilogram to liǎng (liǎng:)
kg to short cwt	Function (linked ; monadic)	kg→short cwt		Convert kilogram to short hundredweight (short cwt:)
kg to short ton	Function (linked ; monadic)	kg→short ton		Convert kilogram to short ton (short ton:)
kg to stone	Function (linked ; monadic)	kg→stone		Convert kilogram to stone (stone:)
kg to ton	Function (linked ; monadic)	kg→ton		Convert kilogram to ton (ton:)
kilo	Command (nonpgm)	*k		Factor 10^3
km to mi.	Function (linked ; monadic)	km→mi.		Convert kilometer to mile (mi.:)
km to nmi	Function (linked ; monadic)	km→nmi		Convert kilometer to nautical mile (nmi:)
km/h to ft/s	Function (linked ; monadic)	km/h→ft/s		Convert kilometer per hour to foot per second (ft/s:)

FullName	Type	Label	Catalog	Description
km/h to fur/ftn	Function (linked ; monadic)	km/h→fur/ftn		Convert kilometer per hour to furlong per fortnight (fur/ftn)
km/h to knot	Function (linked ; monadic)	km/h→knot		Convert kilometer per hour to knot (knot:)
km/h to m/s	Function (linked ; monadic)	km/h→m/s		Convert kilometer per hour to meter per second (m/s:)
km/h to mph	Function (linked ; monadic)	km/h→mph		Convert kilometer per hour to mile per hour (mph:)
km/kWh to kWh/100km	Function (linked ; monadic)	km/kWh→kWh/100km		Convert kilometer per kilowatt-hour to kilowatt-hour per 100 km (kWh/100km:)
km/l to l/100km	Function (linked ; monadic)	km/l→l/100km		Convert kilometer per liter to liter per 100 km (l/100km:)
km/l _e to kWh/100km	Function (linked ; monadic)	km/l _e →kWh/100km		Convert kilometer per liter equivalent to kilowatt-hour per 100 km (kWh/100km:)
km ² to mi ²	Function (linked ; monadic)	km ² →mi ²		Convert square kilometer to square mile (mi ² :)
km ² to nmi ²	Function (linked ; monadic)	km ² →nmi ²		Convert square kilometer to square nautical mile (nmi ² :)
knot to km/h	Function (linked ; monadic)	knot→km/h		Convert knot to kilometer per hour (km/h:)
kWh/100km to km/kWh	Function (linked ; monadic)	kWh/100km→km/kWh		Convert kilowatt-hour per 100 km to kilometer per kilowatt-hour (km/kWh:)
kWh/100km to km/l _e	Function (linked ; monadic)	kWh/100km→km/l _e		Convert kilowatt-hour per 100 km to kilometer per liter equivalent (km/l _e :)
kWh/100km to kWh/100mi	Function (linked ; monadic)	kWh/100km→kWh/100mi		Convert kilowatt-hour per 100 km to kilowatt-hour per 100 mile ()
kWh/100mi to kWh/100km	Function (linked ; monadic)	kWh/100mi→kWh/100km		Convert kilowatt-hour per 100 mile to kilowatt-hour per 100 km (kWh/100km:)
kWh/100mi to mi/kWh	Function (linked ; monadic)	kWh/100mi→mi/kWh		Convert kilowatt-hour per 100 mile to mile per kilowatt-hour (mi/kWh:)
kWh/100mi to mpge _{UK}	Function (linked ; monadic)	kWh/100mi→mpge _{UK}		Convert kilowatt-hour per 100 mile to mpg gasoline equivalent UK (mpge _{UK} :)
kWh/100mi to mpge _{US}	Function (linked ; monadic)	kWh/100mi→mpge _{US}		Convert kilowatt-hour per 100 mile to mpg gasoline equivalent US (mpge _{US} :)
L	Character	L		Character L (76)
L acute	Character	Ĺ		International character Ĺ (313)
L apostrophe	Character	Ĺ		International character Ĺ (317)
l apostrophe lowercase	Character	ĺ		International character ĺ (318)
l lowercase	Character	l		Character l (108)
l stoke lowercase	Character	ł		International character ł (322)
L stroke	Character	Ł		International character Ł (321)
l to gal _{UK}	Function (linked ; monadic)	l→gal _{UK}		Convert liter to UK gallon (gal _{UK} :)
l to gal _{US}	Function (linked ; monadic)	l→gal _{US}		Convert liter to US gallon (gal _{US} :)
l to qt.	Function (linked ; monadic)	l→qt.		Convert liter to quart (qt.:)
l.y. to m	Function (linked ; monadic)	l.y.→m		Convert lightyear to meter (m:)
l/100km to km/l	Function (linked ; monadic)	l/100km→km/l		Convert liter per 100 km to kilometer per liter (km/l:)
l/100km to mpg _{UK}	Function (linked ; monadic)	l/100km→mpg _{UK}		Convert liter per 100 km to mile per gallon UK (mpg _{UK} :)
l/100km to mpg _{US}	Function (linked ; monadic)	l/100km→mpg _{US}		Convert liter per 100 km to mile per gallon US (mpg _{US} :)
Label	Command (PEM)	LBL	LBL	Create local/global label (LBL __ TamLbl(Alpha) menu)
Label exists?	Command	LBL?	LBL?	Test label exists (LBL? __ TamLbl(Alpha) menu)
Laguerre generalised polynomials	Function (triadic)	L _{mα}	L _{mα}	Laguerre generalised polynomials (x = X ; m = Y ; α = Z)
Laguerre polynomials	Function (dyadic)	L _m	L _m	Laguerre polynomials (x = X ; m = Y)
Lambda	Character	Λ		Greek character Λ (923)
lambda lowercase	Character	λ		Greek character λ (955)
Landé's electron g-factor	Constant (#15)	g _e		gfact.elec g _e = -2.00231930436256
Large longint display	Setting	LRG_LI	LRG_LI [•]	Large longint display
Last X	Command	LASTx	LASTx	Recall last X (register L)
lb. to fir	Function (linked ; monadic)	lb.→fir		Convert pound to firkin (fir:)
lb. to kg	Function (linked ; monadic)	lb.→kg		Convert pound to kilogram (kg:)
lbf to N	Function (linked ; monadic)	lbf→N		Convert poundforce to Newton (N:)
lbf-ft to Nm	Function (linked ; monadic)	lbf-ft→Nm		Convert pound-foot to Newton-meter (Nm:)
Leading zeros	Setting	LEAD.0	LEAD.0 []	Leading zeros are on (shortint bases 2, 4, 8, 16) (FF L)
Leading zeros	Flag	LEAD.0		Leading zeros are on (shortint bases 2, 4, 8, 16) (FF L)
Leap year?	Function (monadic)	LEAP?	LEAP?	Test date is in leap year
Least common multiple	Function (dyadic)	LCM	LCM	Least common multiple of X and Y
Left	Symbol	←		Move left (navigation) or arrow character (alpha selection menus) (8592)
Left	Arrow	←		Move left (navigation) or arrow character (alpha selection menus)

FullName	Type	Label	Catalog	Description
Left bit mask	Function (monadic)	MASKL	MASKL	Set left n bits to use as mask (MASKL __ TamNonReg menu)
Left curly bracket	Character	{		Character { (123)
Left French quotation mark	Character	«		Character « (171)
Left justify	Function (monadic)	LJ	LJ	Left justify (within word size) ; returns shift in X and result in Y
Left parenthesis	Symbol	(Open (left) parenthesis (40)
Left square bracket	Character	[Character [(91)
Left-left	Arrow	←		Move all the way to the left (navigation) or arrow character (alpha selection menus)
Left-right arrows	Character	↔		Character ↔ (8644)
Legendre polynomials	Function (dyadic)	P_n	P_n	Legendre polynomials ($x = X$; $n = Y$)
Less or equal than	Character	≤		Character ≤ (8804)
Less than	Character	<		Character < (60)
lī to m	Function (linked ; monadic)	lī→m		Convert lī to meter (m)
liång to kg	Function (linked ; monadic)	liång→kg		Convert liång to kilogram (kg.)
Linear fit	Setting	LinF	LinF [.]	Linear curve fitting
Linear interpolation	Function (triadic)	LINPOL	LINPOL	Linear interpolation ; also works for complex numbers (<from> ENTER <towards> ENTER <at> (fraction))
Linear Regression	Command	L.R.	L.R.	Linear Regression
List graph coordinates	Command (deprecated)	LISTXY	LISTXY	List the actual STATS graph coordinates (7 digit floating point)
LN(1+x)	Function (monadic)	LN(1+x)	LN(1+x)	More accurate calculation of LN(1+x) for $x \approx 0$
LNβ	Function (dyadic)	LNβ	LNβ	Natural logarithm of Euler's Beta function
LNΓ	Function (monadic)	LNΓ	LNΓ	Natural logarithm of the Gamma function
Load full backup	Command (nonpgm)	LOAD	LOAD	Load full backup from SAVE file C47.sav in FAT and auto-clear user mode (SAVFILES ; inverse : SAVE)
Load program	Command (nonpgm)	LOADP	LOADP	Load program from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)
Load program	Command	X.LOAD	X.LOAD __ TamNonReg menu)	Load XEQC program (X.LOAD __ TamNonReg menu)
Load registers	Command (nonpgm)	LOADR	LOADR	Load registers from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)
Load Sigma registers	Command (nonpgm)	LOADΣ	LOADΣ	Load Sigma registers from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)
Load state file	Command (nonpgm)	LOADST	LOADST	Load state file from SAVEST file <state.s47> in FAT (File open dialog (STATE) ; Inverse : SAVEST)
Load system state	Command (nonpgm)	LOADSS	LOADSS	Load system state from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)
Load variables	Command (nonpgm)	LOADV	LOADV	Load variables from SAVE file C47.sav in FAT (SAVFILES ; inverse : SAVE)
Lock normal script	Command (internal)	SCRNRM		Lock normal script
Lock subscript	Command (internal)	SCRSUB		Lock subscript
Lock superscript	Command (internal)	SCRSUP		Lock superscript
Log normal (inverse)	Function (tbd)	LgNrm ⁻¹	LgNrm ⁻¹	Log normal probability inverse function
Log normal cdf (lower)	Function (tbd)	LgNrm _▲	LgNrm _▲	Log normal cumulative distribution (lower tail)
Log normal cdf (upper)	Function (tbd)	LgNrm _▲	LgNrm _▲	Log normal cumulative distribution (upper tail)
Log normal distribution	MENU	LgNrm:	LgNrm:	Log normal probability distribution (RegI = μ = mean ; RegJ = σ = standard deviation)
Log normal pdf	Function (tbd)	LgNrm _p	LgNrm _p	Log normal probability density function
LOG _x y	Function (dyadic)	LOG _x y	LOG _x y	Logarithm of Y for base X
Logarithmic fit	Setting	LogF	LogF [.]	Logarithmic curve fitting
Logistic (inverse)	Function (tbd)	Logis _p	Logis _p	Logistic probability inverse function
Logistic cdf (lower)	Function (tbd)	Logis ⁻¹	Logis ⁻¹	Logistic cumulative distribution (lower tail)
Logistic cdf (upper)	Function (tbd)	Logis _▲	Logis _▲	Logistic cumulative distribution (upper tail)
Logistic distribution	MENU	Logis:	Logis:	Logistic probability distribution (RegI = μ = location ; RegJ = s = scale)
Logistic pdf	Function (tbd)	Logis _▲	Logis _▲	Logistic probability density function
Long integer	Function (monadic)	LINT	LINT	Convert to long integer (max 1000 digits) (#ENTER)
Longint variables	MENU	L.INTS	L.INTS	Auto-generated catalog of variables of the specified type: long integer
Looping	MENU	LOOP	LOOP	Looping (programming) functions
Low Battery	Flag (system)	LOWBAT		Low battery voltage

FullName	Type	Label	Catalog	Description
Low bin	Command	↓BIN	↓BIN	Low bin (nBINS : ; ↓BIN : ; ↑BIN : (3 stack levels))
Lower limit of integration	Variable (real)	↓Lim	↓Lim	Lower limit of integration (reserved real variable) (↓Lim :)
Lower quantile	Command	× _{Q1}	× _{Q1}	Lower quantile for both X and Y (Q _{1 x} ; Q _{1 y} = (2 stack levels))
LU decomposition	Function (monadic)	M.LU	M.LU	Decompose matrix into lower (L) in Y and upper (U or R) matrix in X ; pivot matrix in Z (LU (LR) factorisation)
M	Character	M		Character M (77)
m lowercase	Character	m		Character m (109)
m to au	Function (linked ; monadic)	m→au		Convert meter to astronomical unit (au:)
m to brds	Function (linked ; monadic)	m→brds		Convert meter to beardsecond (brds:)
m to chī	Function (linked ; monadic)	m→chī		Convert meter to chī (chī:)
m to cùn	Function (linked ; monadic)	m→cùn		Convert meter to cùn (cùn:)
m to fathom	Function (linked ; monadic)	m→fathom		Convert meter to fathom (fathom:)
m to fēn	Function (linked ; monadic)	m→fēn		Convert meter to fēn (fēn:)
m to ft.	Function (linked ; monadic)	m→ft.		Convert meter to foot (ft:)
m to fur	Function (linked ; monadic)	m→fur		Convert meter to furlong (fur:)
m to l.y.	Function (linked ; monadic)	m→l.y.		Convert meter to lightyear (l.y:)
m to lǐ	Function (linked ; monadic)	m→lǐ		Convert meter to lǐ (lǐ:)
m to mi.	Function (linked ; monadic)	m→mi.		Convert meter to mile (mi:)
m to nmi	Function (linked ; monadic)	m→nmi		Convert meter to nautical mile (nmi:)
m to pc	Function (linked ; monadic)	m→pc		Convert meter to parsec (pc:)
m to surveyft.us	Function (linked ; monadic)	m→surveyft.us		Convert meter to US survey foot (surveyft.us:)
m to yd.	Function (linked ; monadic)	m→yd.		Convert meter to yard (yd:)
m to yīn	Function (linked ; monadic)	m→yīn		Convert meter to yīn (yīn:)
m to zhàng	Function (linked ; monadic)	m→zhàng		Convert meter to zhàng (zhàng:)
m/s to ft/s	Function (linked ; monadic)	m/s→ft/s		Convert meter per second to foot per second (ft/s:)
m/s to fur/ftn	Function (linked ; monadic)	m/s→fur/ftn		Convert meter per second to furlong per fortnight (fur/ftn:)
m/s to km/h	Function (linked ; monadic)	m/s→km/h		Convert meter per second to kilometer per hour (km/h:)
m/s to mph	Function (linked ; monadic)	m/s→mph		Convert meter per second to mile per hour (mph:)
Magnetic flux quantum	Constant (#74)	φ ₀		fluxq.magn φ ₀ = +2,067833848461929323081115412147497 × 10 ⁻¹⁵ (Vs)
Magnitude	Function (monadic)	x	x	Magnitude (absolute value) of complex number
Mantissa	Function (monadic)	MANT	MANT	Mantissa
Mass conversion	MENU	Mass:	Mass:	Convert between units of mass
Mass of the Earth	Constant (#34)	M _⊕		mass.earth M _⊕ = +5.9736 × 10 ²⁴ (kg)
Mass of the Moon	Constant (#24)	M _{Moon}		mass.moon M _{Moon} = +7.349 × 10 ²² (kg)
Mass of the Sun	Constant (#33)	M _⊙		mass.sun M _⊙ = +1.9891 × 10 ³⁰ (kg)
Math keys full cycle	Setting	M.1234		Longpress control : full Math keys longpress cycle
Math keys skip fg	Setting	M.14		Longpress control : skip f and g Math keys longpress cycle
Math keys skip g	Setting	M.124		Longpress control : skip g Math keys longpress cycle
Matrix	MENU	MATX	MATX	Matrix functions
Matrix A	Variable (matrix)	Mat A	Mat_A	Reserved matrix variable Matrix A (Mat_A) (Enter data using menu SIM EQ)
Matrix B	Variable (matrix)	Mat B	Mat_B	Reserved matrix variable Matrix B (Mat_B) (Enter data using menu SIM EQ)
Matrix dimension (X)	Function (monadic)	DIM?	M.DIM?	Dimensions of matrix in X
Matrix editor	MENU	EDIT	M.EDIT	Matrix edit functions (MIM = Matrix Input Mode)
Matrix simultaneous equations	MENU (item)	SIM EQ	SIM_EQ	Matrix simultaneous equations functions (Mat A • Mat X = Mat B) (SIM_EQ __ TamNonReg menu ; CAT.MENUS M.SIMQ)
Matrix simultaneous equations	MENU	SIM EQ	M.SIMQ	Matrix simultaneous equations functions (Mat A • Mat X = Mat B)
Matrix variables	MENU	MATRS	MATRS	Auto-generated catalog of variables of the specified type: matrix
Matrix X	Command	Mat X	Mat_X	Create reserved matrix variable Matrix X (Mat_X) and solve simultaneous equations

FullName	Type	Label	Catalog	Description
Matrix X	Variable (matrix)	Mat_X	Mat_X	Reserved matrix variable Matrix X (Mat_X) (Solve equations using menu SIM EQ)
Matrix?	Function (monadic)	MATR?	MATR?	Test X is a matrix
Maxima	Command	X _{MAX}	X _{MAX}	Maxima (X _{MAX} ; Y _{MAX} = (2 stack levels))
Maximum	Function (dyadic)	max	max	Maximum of X and Y
Maximum denominator	Variable (longint)	D.MAX	D.MAX	Maximum denominator (system long integer variable, write protected)
Mean radius of the Earth	Constant (#44)	R _⊕		rad.earth R _⊕ = +6.37101 × 10 ⁶ (m)
Mean radius of the Moon	Constant (#41)	R _{Moon}		rad.moon R _{Moon} = +1.73753 × 10 ⁶ (m)
Mean radius of the Sun	Constant (#43)	R _☉		rad.sun R _☉ = +6.96 × 10 ⁸ (m)
Mean rate of return	Command	%MRR	%MRR	Mean rate of return in percentage per period (% :)
Measured angle	Character	∠		Character ∠ (8737)
Median	Command	X _{MEDN}	X _{MEDN}	Sort the data and return the middle value for both X and Y ; for an even number of samples, the arithmetic mean of the two middle values is returned (md _x ; md _y = (2 stack levels))
Median absolute deviation	Command	X _{MAD}	X _{MAD}	Median absolute deviation for both X and Y ; this is the median of the differences between each data point and the overall median (mad _x ; mad _y = (2 stack levels))
Mega	Command (nonpgm)	•M		Factor 10 ⁶
Memory (RAM)	Command	MEM?	MEM?	Amount of free RAM memory
Menu fg-highlighting Full	Setting	fg.FUL	fg.FUL (•)	Show full horizontal f- and g-lines indicating the state of the f- or g-shift in menu
Menu fg-highlighting Limited	Setting	fg.LIM	fg.LIM ()	Show limited horizontal f- and g-lines indicating the state of the f- or g-shift in menu
Menu fg-highlighting Off	Setting	fg.OFF	fg.OFF ()	Do not show horizontal f- and g-lines indicating the state of the f- or g-shift in menu
Menu variable	Command (PEM)	MVAR	MVAR	Define menu variable for VarMNU (MVAR __ Tam menu)
MENUS	MENU (ASM)	MENUS		Catalog of all menus (including user defined menus) (Type characters 1-2 to search)
Message	Command	MSG	MSG	Show error message (MSG __ Tam menu)
mi. to km	Function (linked ; monadic)	mi.→km		Convert mile to kilometer (km:)
mi. to m	Function (linked ; monadic)	mi.→m		Convert mile to meter (m:)
mi. to nmi	Function (linked ; monadic)	mi.→nmi		Convert mile to nautical mile (nmi:)
mi/kWh to kWh/100mi	Function (linked ; monadic)	mi/kWh→kWh/100mi		Convert mile per kilowatt-hour to kilowatt-hour per 100 mile (kWh/100mi:)
micro	Command (nonpgm)	•μ		Factor 10 ⁻⁶
m ² to ha	Function (linked ; monadic)	m ² →ha		Convert square meter to hectare (ha:)
m ² to mü	Function (linked ; monadic)	m ² →mü		Convert square meter to mü (mü:)
mi ² to km ²	Function (linked ; monadic)	mi ² →km ²		Convert square mile to square kilometer (km ² :)
Mileage conversion	MENU	Ymmv:	Ymmv:	Convert between units of mileage ("Your mileage may vary") ("E" designates "kWh")
milli	Command (nonpgm)	•m		Factor 10 ⁻³
Minima	Command	X _{MIN}	X _{MIN}	Minima (X _{MIN} ; Y _{MIN} = (2 stack levels))
Minimum	Function (dyadic)	min	min	Minimum of X and Y
Minus	Function (dyadic)	-		Minus (45)
Minus infinity	Constant (#76)	-∞		inf.minus = -∞
Minutes	Function (monadic)	MIN	MIN	Minutes (of time)
Minutes & seconds	Function (cyclic ; monadic)	.ms	.ms	Convert sexagesimal format input sequence or decimal stack value to hh:mm:ss hours or dd°mm'ss" degrees (cyclic) (NIM input treated as sexagesimal (hh/dd.mmss) format ; stack input treated as decimal value)
Mirror bits	Function (monadic)	MIRROR	MIRROR	Flip bits
Miscellaneous conversions	MENU	Misc:	Misc:	Time, temperature, torque, power and field ratio conversions
m ³ to barrel	Function (linked ; monadic)	m ³ →barrel		Convert cubic meter to barrel (barrel:)
ml to floz _{UK}	Function (linked ; monadic)	ml→floz _{UK}		Convert milliliter to UK fluid ounce (floz _{UK} :)
ml to floz _{US}	Function (linked ; monadic)	ml→floz _{US}		Convert milliliter to US fluid ounce (floz _{US} :)
mm to in.	Function (linked ; monadic)	mm→in.		Convert millimeter to inch (in.:)
mm to point	Function (linked ; monadic)	mm→point		Convert millimeter to point (point:)
mm.Hg to Pa	Function (linked ; monadic)	mm.Hg→Pa		Convert millimeter of Mercury to Pascal (Pa:)

FullName	Type	Label	Catalog	Description
Mode settings	MENU	MODE	MODE	System (mode) settings with status indication and modification
Model	MENU	MODEL	MODEL	Model functions
Modulo	Function (dyadic)	MOD	MOD	Y modulo X
Molar gas constant	Constant (#38)	R		c.mol.gas R = +8.31446261815324 (J/mol K)
Month	Function (monadic)	MONTH	MONTH	Month (of date)
Month day year	Setting (pgm)	MDY	MDY ()	Date display mode MM/DD/YYYY (MM/DD/YYYY)
Month day year	Flag (system)	MDY		Date display mode MM/DD/YYYY (MM/DD/YYYY)
Moon orbit (semi major axis)	Constant (#2)	a _{Moon}		orb.moon a _{Moon} = +3.844 × 10 ⁸ (m)
More programming functions	MENU	P.FN...	P.FN...	More programming functions
More trig/hyperbolics	MENU (47)	TRG...	TRG _{F47} ...	Extended trigonometry (and access to hyperbolic) functions
mpge _{UK} to kWh/100mi	Function (linked ; monadic)	mpge _{UK} →kWh/100m		Convert mpg gasoline equivalent UK to kilowatt-hour per 100 mile (kWh/100mi:)
mpge _{US} to kWh/100mi	Function (linked ; monadic)	mpge _{US} →kWh/100m		Convert mpg gasoline equivalent US to kilowatt-hour per 100 mile (kWh/100mi:)
mpg _{UK} to l/100km	Function (linked ; monadic)	mpg _{UK} →l/100km		Convert mile per gallon UK to liter per 100 km (l/100km:)
mpg _{US} to l/100km	Function (linked ; monadic)	mpg _{US} →l/100km		Convert mile per gallon US to liter per 100 km (l/100km:)
mph to fur/ftn	Function (linked ; monadic)	mph→fur/ftn		Convert mile per hour to furlong per fortnight (fur/ftn:)
mph to km/h	Function (linked ; monadic)	mph→km/h		Convert mile per hour to kilometer per hour (km/h:)
mph to m/s	Function (linked ; monadic)	mph→m/s		Convert mile per hour to meter per second (m/s:)
Mu	Character	Μ		Greek character M (924)
mu lowercase	Character	μ		Greek character μ (956)
mū to m ²	Function (linked ; monadic)	mū→m ²		Convert mū to square meter (m ² :)
Multiplication symbol	Flag	MULT×		Set for multiplication symbol × for exponential and complex notation, clear for •
Multiplication symbol ×	Setting (pgm)	MULT×	MULT× (•)	Multiplication symbol × for exponential (and for complex numbers if CPXmul is set)
Multiplication symbol •	Setting (pgm)	MULT•	MULT• ()	Multiplication symbol • for exponential (and for complex numbers if CPXmul is set)
Multiply	Function (dyadic)	[×]	×	Multiply Y by X (215)
Multiply into	Function (monadic)	ST0×	ST0×	Multiply register or variable by X (ST0× __ TamStoRcl(Alpha) menu)
Muon magnetic moment	Constant (#71)	μ _B		mgmom.muon μ _B = -4.4904483 × 10 ⁻²⁶ (J/T)
Muon rest mass	Constant (#32)	m _μ		mass.muon m _μ = +1.883531627 × 10 ⁻²⁸ (kg)
MVAR	MENU	MVAR		(Internal) menu MVAR for VarMNU
MyAlpha	MENU	Myα	Myα	MyAlpha is where to assign special characters for easy entry (Longpress AIM [EXIT])
MyAlpha reset	Command	A.RESET		Reset MyAlpha user soft button assignments
MyMenu	MENU	MyM	MyMenu	MyMenu is where to assign user selected functions or user selected or defined menus ; two predefined options settable in menu KEYS (Documentation page 1 shows option M.ENG ; page 2 shows option M.FIN ; Hidden : Longpress [EXIT])
MyMenu ENG	Command	M.ENG		Populate MyMenu with engineering functions (Documented as MyMenu page 1 (default))
MyMenu FIN	Command	M.FIN		Populate MyMenu with financial functions (Documented as MyMenu page 2)
MyMenu reset	Command	M.RESET		Reset MyMenu user soft button assignments
MyMenu shown	Setting	MyM	MyM [•]	Base MyMenu shown (when all menus are exited using EXIT)
N	Character	N		Character N (78)
n	Command	n	nΣ	Number of samples
N acute	Character	Ñ		International character Ñ (323)
n acute lowercase	Character	ñ		International character ñ (324)
N caron	Character	Ň		International character Ň (327)
n caron lowercase	Character	ň		International character ň (328)
n lowercase	Character	n		Character n (110)
N tilde	Character	Ñ		International character Ñ (209)
n tilde lowercase	Character	ñ		International character ñ (241)
N to lbf	Function (linked ; monadic)	N→lbf		Convert Newton to poundforce (lbf:)
N47 keyboard layout	Layout (SIM)	N47		N47: Exp 2 shift L (DM32 mould) /x+ R Up Dn top ; in progress ; N = new
Name equation	Symbol	:		Used in equation editor to name an equation (NAME : <equation>) (58)
nano	Command (nonpgm)	•n		Factor 10 ⁻⁹

FullName	Type	Label	Catalog	Description
Natural logarithm	Function (monadic)	LN	LN	Natural logarithm (base e)
Natural logarithm	Function (monadic)	LN		Natural logarithm (base e)
Negative binomial (inverse)	Function (tbd)	NBin ⁻¹	NBin ⁻¹	Negative binomial probability inverse function
Negative binomial cdf (lower)	Function (tbd)	NBin _⋮	NBin _⋮	Negative binomial cumulative distribution (lower tail)
Negative binomial cdf (upper)	Function (tbd)	NBin _⋮	NBin _⋮	Negative binomial cumulative distribution (upper tail)
Negative binomial distribution	MENU	NBin:	NBin:	Negative binomial probability distribution (RegI = p = probability ; RegJ = n = number of samples)
Negative binomial pdf	Function (tbd)	NBin _p	NBin _p	Negative binomial probability density function
Negative exclamation mark	Character	⓪		Character ⓪ (9263)
Neighbour	Function (dyadic)	NEIGHB	NEIGHB	Neighbour value of X compared to Y (next integer or next machine representable real)
Neutron / proton rest mass	Constant (#26)	m _n /m _p		r.neu.prot m _n /m _p = +1.00137841898
Neutron magnetic moment	Constant (#68)	μ _n		magnom.neu μ _n = -9.662365 × 10 ⁻²⁷ (J/T)
Neutron rest mass	Constant (#25)	m _n		mass.neu m _n = +1.67492749804 × 10 ⁻²⁷ (kg)
New matrix	Command (submnu)	NEW	M.NEW	Create new matrix (MIM = Matrix Input Mode)
New program	Command	X.NEW		New XEQC program
Newtonian constant of gravitation	Constant (#12)	G		c.grav.nwt G = +6.6743 × 10 ⁻¹¹ (m ³ /kg.s ²)
Newtonian constant of gravitation	Constant (#56)	γ		c.grav.nwt γ = +6.6743 × 10 ⁻¹¹ (m ³ /kg.s ²)
Next fit	Command (cyclic ; nonpgm)	NXTFIT		Assess next curve fitting (cyclic)
Next prime	Function (monadic)	NEXTP	NEXTP	Next prime number
Nm to lbf-ft	Function (linked ; monadic)	Nm→lbf-ft		Convert Newton-meter to pound-foot (lbf-ft)
nmi to km	Function (linked ; monadic)	nmi→km		Convert nautical mile to kilometer (km)
nmi to m	Function (linked ; monadic)	nmi→m		Convert nautical mile to meter (m)
nmi to mi.	Function (linked ; monadic)	nmi→mi.		Convert nautical mile to mile (mi.)
nmi ² to km ²	Function (linked ; monadic)	nmi ² →km ²		Convert square nautical mile to square kilometer (km ²)
No FP separator	Setting (pgm)	NONE	FNONE ()	No fractional part separator (Menu shows symbol ⓪)
No IP separator	Setting (pgm)	NONE	INONE ()	No integer part separator (Menu shows symbol ⓪)
No operation	Command (PEM ; deprecated)	NOP	NOP	No operation (empty step)
Nominal mean angular velocity of the Earth	Constant (#75)	ω		vangl.earth ω = +7.292115 × 10 ⁻⁵ (rad/s)
NOR	Function (dyadic)	NOR	NOR	Logical not OR (bitwise)
Normal (inverse)	Function (tbd)	Norml ⁻¹	Norml ⁻¹	Normal probability inverse function
Normal cdf (lower)	Function (tbd)	Norml _⋮	Norml _⋮	Normal cumulative distribution (lower tail)
Normal cdf (upper)	Function (tbd)	Norml _⋮	Norml _⋮	Normal cumulative distribution (upper tail)
Normal distribution	MENU	Norml:	Norml:	Normal probability distribution (RegI = μ = mean ; RegJ = σ = standard deviation)
Normal pdf	Function (tbd)	Norml _p	Norml _p	Normal probability density function
Normal results	Command	SPCRES0	SPCRES0	Do not allow special results of calculations (infinity, not-a-number) ; an error will occur for such events (For programming purposes)
NOT	Function (monadic)	NOT	NOT	Logical NOT
Not	Character	¬		Character ¬ (172)
Not a Number	Constant (#36)	NaN		not.a.nr NaN = Not a number
Not a number?	Function (monadic)	NaN?	NaN?	Test X is Not-a-Number
Not AND	Function (dyadic)	NAND	NAND	Logical not AND (bitwise)
Not equal to	Character	≠		Character ≠ (8800)
Nu	Character	Ν		Greek character N (925)
nu lowercase	Character	ν		Greek character ν (957)
Nuclear magneton	Constant (#70)	μ _n		magn.nucl μ _n = +5.0507837461 × 10 ⁻²⁷ (J/T)
Number (base)	Setting (pgm ; stack)	#	→INT	Set number base ; operates on all stack registers depending on BASE _{HP} ; reset by [.d] (gShifted [LOG]) (Indirection (→) activates TamNonRegInd ; Shortcuts : H:16 (hex) ; ENTER or D:10 (shortint); O:8 (octal) ; B:2 (binary) ; Info : SBI depends on SBfrac)

FullName	Type	Label	Catalog	Description
Number (base)	Setting (pgm ; stack)	→INT	→INT	Set number base ; operates on all stack registers depending on BASE _{HP} ; reset by [,d] (gShifted [LOG]) (Indirection (→) activates TamNonRegInd ; Shortcuts : H:16 (hex) ; ENTER or D:10 (shortint); O:8 (octal) ; B:2 (binary) ; Info : SBI depends on SBfrac)
Number base	MENU	BASE	BASE	Number base operations (shortint) (X: hexadecimal ; X: binary)
Number of bins	Command	nBINS		Number of bins in histogram (nBINS ; ; ¶BIN ; ; ¶BIN : (3 stack levels))
Number of bits set	Function (monadic)	#B	#B	Count number of bits set
Number of local registers	Command	LocR?	LocR?	Number of local registers (current routine) ; set by LocR
Numeric Entry	Flag	NUM.IN		Numeric entry active
Numlock	Setting	NUM		Lock numeric alpha input
Numlock (lower)	Command (internal)	NLock		Lock numeric alpha input with capslock off
Numlock (upper)	Command (internal)	Nulock		Lock numeric alpha input with capslock on
0	Character	0		Character 0 (79)
0 acute	Character	ó		International character ó (211)
o acute lowercase	Character	ó		International character ó (243)
0 breve	Character	ö		International character ö (334)
o breve lowercase	Character	ö		International character ö (335)
0 circumflex	Character	ô		International character ô (212)
o circumflex lowercase	Character	ô		International character ô (244)
0 diaeresis	Character	ö		International character ö (214)
o diaeresis lowercase	Character	ö		International character ö (246)
0 grave	Character	ò		International character ò (210)
o grave lowercase	Character	ò		International character ò (242)
o lowercase	Character	o		Character o (111)
0 macron	Character	ō		International character ō (332)
o macron lowercase	Character	ō		International character ō (333)
0 stroke	Character	ø		International character ø (216)
o stroke lowercase	Character	ø		International character ø (248)
0 tilde	Character	õ		International character õ (213)
o tilde lowercase	Character	õ		International character õ (245)
Obelus	Character	÷		Character ÷ (247)
Octal	Setting (pgm)	OCT	OCT	Convert X to octal and toggle octal mode
Octal base	Shortcut (TAM)	#8		TAM shortcut for setting octal base (Shortcut O(ctal) (C47.44.41))
Odd?	Function (monadic)	ODD?	ODD?	Test X is integer AND odd
OE	Character	œ		International character œ (338)
oe lowercase	Character	œ		International character œ (339)
Off	Command	OFF	OFF	Turn off calculator
Old (matrix)	Command	OLD	M.OLD	Revert to old element (while editing)
Omega	Character	Ω		Greek character Ω (937)
omega lowercase	Character	ω		Greek character ω (969)
omega tonos lowercase	Character	ώ		Greek character ώ (974)
Omicron	Character	0		Greek character 0 (927)
omicron lowercase	Character	o		Greek character o (959)
omicron tonos lowercase	Character	ó		Greek character ó (972)
Operator a	Command	a	op_a	Insert value of $1 \angle 120^\circ$
Operator a ²	Command	a ²	op_a ²	Insert value of $1 \angle 240^\circ$
OR	Function (dyadic)	OR	OR	Logical OR
Or	Character	∨		Character ∨ (8744)
Orthogonal	MENU	Orthog	ORTHOG	Orthogonal polynomials
Orthogonal fit	Setting	OrthoF	OrthoF []	Orthogonal curve fitting

FullName	Type	Label	Catalog	Description
Overflow	Flag	OVERFL		Status of overflow bit (FF B ; Info : SBI depends on SBoc)
oz to g	Function (linked ; monadic)	oz→g		Convert ounce to gram (g:)
P	Character	P		Character P (80)
p lowercase	Character	p		Character p (112)
Pa to atm	Function (linked ; monadic)	Pa→atm		Convert Pascal to atmosphere (atm:)
Pa to bar	Function (linked ; monadic)	Pa→bar		Convert Pascal to bar (bar:)
Pa to in.Hg	Function (linked ; monadic)	Pa→in.Hg		Convert Pascal to inch of Mercury (in.Hg:)
Pa to mm.Hg	Function (linked ; monadic)	Pa→mm.Hg		Convert Pascal to millimeter of Mercury (mm.Hg:)
Pa to psi	Function (linked ; monadic)	Pa→psi		Convert Pascal to pounds per square inch (psi:)
Pa to torr	Function (linked ; monadic)	Pa→torr		Convert Pascal to torr (torr:)
Parabolic fit	Setting	ParabF	ParabF []	Parabolic curve fitting
Parallel	Function (dyadic)			Parallel impedance = $(X \times Y) / (X + Y)$
Pause	Command (PEM)	PAUSE	PAUSE	Pause program for n ticks (0-99 ; one tick is 10 ms) ; continues after delay or on keypress (PAUSE __ TamNonReg menu)
Payment	Variable (real)	PMT	PMT	Payment (reserved real variable)
pc to m	Function (linked ; monadic)	pc→m		Convert parsec to meter (m:)
Pct of sum and Delta pct to mean	Function (monadic)	%Σ,Δ% \bar{x}	%Σ,Δ% \bar{x}	Percentage of x to Σx and Delta percentage to mean using statistics matrix (STATS) (% ; Δ% : (2 stack levels))
Percent	Function (dyadic)	%	%	X Percent of Y, keeping Y on stack
Percent	Character	%		Character % (37)
Percentage of sum	Function (monadic)	%Σ	%Σ	Percentage of x to Σx (% :)
Percentage of total	Function (dyadic)	%T	%T	Percentage of total, keeping Y on stack (% :)
Percentile	Command	x%ILE	x%ILE	Percentile (pctile _x ; pctile _y = (2 stack levels))
Period	Character	.		Character . (46)
Permutations	Function (dyadic)	P _{y,x}	PERM	Permutations of X out of Y
Perpendicular	Character	⊥		Character ⊥ (8869)
Peta	Command (nonpgm)	•P		Factor 10 ¹⁵
Pgm running	SBI	P		Program running
Pgm waiting	SBI	⓪		Program waiting (user input required)
Phi	Character	φ		Greek character φ (934)
phi lowercase	Character	φ		Greek character φ (966)
pi	Command	π		Insert value of pi
Pi	Character	Π		Greek character Π (928)
pi lowercase	Character	π		Greek character π (960)
pico	Command (nonpgm)	•p		Factor 10 ⁻¹²
Pipe	Character			Character (124)
Planck / 2pi	Character	ħ		Character ħ (8463)
Planck constant	Constant (#18)	h		c.planck h = +6.62607015 × 10 ⁻³⁴ (Js)
Planck length	Constant (#22)	l _{PL}		len.planck l _{PL} = +1.616255 × 10 ⁻³⁵ (m)
Planck mass	Constant (#28)	m _{PL}		mass.planck m _{PL} = +2.176435 × 10 ⁻⁸ (kg)
Planck temperature	Constant (#51)	T _P		temp.planck T _P = +1.416785 × 10 ³² (K)
Planck time	Constant (#52)	t _{PL}		time.planck t _{PL} = +5.391245 × 10 ⁻⁴⁴ (s)
Play sounds	Command	PLAY	PLAY	Play sounds (input from n×3 or n×2 matrix variable having rows : [frequency, duration, volume] with frequency in Hz (0 = silent) ; duration in ms (max 2000) ; volume) ; volume element is optional (PLAY _ TamLabel menu)
Plot statistics	MENU (item)	PLSTAT	PLSTAT	Plot statistics (PLOTMNU)
Plot statistics	MENU	PLSTAT	PLOTMNU	Plot statistics (Regular label: PLSTAT)
Plotting	MENU	PLOT	PLOT	Plotting and summation functions
Plus	Function (dyadic)	+		Plus (43)

FullName	Type	Label	Catalog	Description
Plus infinity	Constant (#77)	$+\infty$		inf.plus = $+\infty$
Plus-minus	Character	\pm		Character \pm (177)
point to mm	Function (linked ; monadic)	point \rightarrow mm		Convert point to millimeter (mm:)
Poisson (inverse)	Function (tbd)	Poiss _p	Poiss _p	Poisson probability inverse function
Poisson cdf (lower)	Function (tbd)	Poiss ⁻¹	Poiss ⁻¹	Poisson cumulative distribution (lower tail)
Poisson cdf (upper)	Function (tbd)	Poiss _u	Poiss _u	Poisson cumulative distribution (upper tail)
Poisson distribution	MENU	Poiss:	Poiss:	Poisson probability distribution (RegI = λ = expected rate of events)
Poisson pdf	Function (tbd)	Poiss _d	Poiss _d	Poisson probability density function
Polar	Setting (pgm)	POLAR	POLAR ()	Polar representation of complex numbers (internal value is RECT) (FF X (TAM) ; Info : SBI depends on SBcpx)
Polar	Flag	POLAR		Set for polar representation of complex numbers, clear for rectangular display (FF X (TAM) ; Info : SBI depends on SBcpx)
Polynomial 3rd degree	Command	cubic		Create polynomial 3rd degree ($b3 \times x^3 + b2 \times x^2 + b1 \times x + b0$)
Pop local registers	Command (PEM)	PopLR	PopLR	Pop local registers (no return to calling routine)
Population standard deviation	Command	σ	σ	Population standard deviation ($\sigma_x ; \sigma_y = (2 \text{ stack levels})$)
Pound	Character	£		Character £ (163)
Power conversion	MENU	Power:	Power:	Convert between units of power
Power fit	Setting	PowerF	PowerF []	Power curve fitting
Precision	Command	S _{mi}	S _{mi}	Precision of measuring instrument investigated, requires 30 data pairs (s _{mi} =)
Present value	Variable (real)	PV	PV	Present value (reserved real variable)
Prime?	Function (monadic)	PRIME?	PRIME?	Test absolute value of integer part of X is prime
Print	Command	PRN	PRN	Print
Print all registers	Command	▣ALLr	▣ALLr	Print all registers - save contents as text to data file in FAT (Filename format YYYYMMDD-HHMMSS00.REGS)
Print byte	Command (strike)	▣#	▣#	Print single byte
Print character	Command (strike)	▣CHAR	▣CHAR	Print single character
Print column	Command (strike)	▣TAB	▣TAB	Position print head to column
Print LCD	Command (strike)	▣LCD	▣LCD	Print screen contents
Print program	Command	▣PROG	▣PROG	Print current program (to text file in FAT) (▣PROG _ TamLabel ; DMCP : File save dialog (PROGRAMS))
Print register	Command (strike)	▣r	▣r	Print register
Print registers	Command (strike)	▣REGS	▣REGS	Print registers (sss.nn means print registers from sss through sss + nn - 1)
Print stack	Command	▣STK	▣STK	Print stack registers - save contents as text to data file in FAT (Filename format YYYYMMDD-HHMMSS00.REGS)
Print sums	Command (strike)	▣Σ	▣Σ	Print summation registers
Print user items	Command (strike)	▣USER	▣USER	Print user items: variable names and programs
Print width	Command (strike)	▣WIDTH	▣WIDTH	Number of print columns or pixels (depending on ▣MODE)
Print X	Command (strike)	▣x	▣x	Print stack register X
Printer	Character	▣		Character ▣ (9113)
Printer Active	Flag	PRACT		Printing is enabled
Printer advance	Command (strike)	▣ADV	▣ADV	Print buffer followed by linefeed (advance)
Printer delay	Command (strike)	▣DLAY	▣DLAY	Set printer delay
Printer mode	Command (strike)	▣MODE	▣MODE	Set printer mode (0: printer font ; 1: variable ; 2: small ; 3: ascii)
Printer off	Command	PROFF	PROFF	Set printer off
Printer on	Command	PRON	PRON	Set printer on
Printer on/off	Setting	PRNTR	PRNTR []	Set printer on/off
Printing	SBI	▣		Printing
Printing	MENU	PRINT	PRINT	Printing functions
Printing	Flag (system)	PRINTS		Calculator is sending data to printer (SBI depends on SBprn)

FullName	Type	Label	Catalog	Description
Probability	MENU	PROB	PROB	Probability functions
Product (programmable)	Command	Π_n	Π_n	Real or complex product using specified program, with iteration counter, interrupt by keypress (Π_n __ TamLbl(Alpha) menu ; <from> ENTER <to> ENTER <step>)
PROG	MENU (TAM ; ASM)	PROG	PROG	Presented in TAM menus for commands accessing labels (CAT.PROGS.* menu ; Type characters 1-2 to search)
Program begin	Character	$\bar{\uparrow}$		Character $\bar{\uparrow}$ (9259)
Program for integrator	Command	PGMINT	PGMINT	The program to be used by the integrator (PGMINT __ TamLbl(Alpha) menu)
Program for solver	Command	PGMSLV	PGMSLV	The program to be used by the solver (PGMSLV __ TamLbl(Alpha) menu)
Programming	Command	PRGM		Enter programming mode (PEM = Program Entry Mode ; starts UPPERCASE)
Programming functions	MENU	P.FN	P.FN	Programming functions
Programs	MENU (deprecated)	PROGS	PROGS	Auto-generated catalog of programs
Proper Fractions	Flag	PROPFR		Proper fractions are used (or improper)
Proton / electron rest mass	Constant (#29)	m_p/m_e		r.prot.elec $m_p/m_e = +1.83615267343 \times 10^3$
Proton gyromagnetic ratio	Constant (#58)	γ_p		r.gyro.prot $\gamma_p = +2.6752218744 \times 10^8$
Proton magnetic moment	Constant (#69)	μ_p		mgmom.prot $\mu_p = +1.41060679736 \times 10^{-26}$ (J/T)
Proton rest mass	Constant (#27)	m_p		mass.prot $m_p = +1.67262192369 \times 10^{-27}$ (kg)
Psi	Character	Ψ		Greek character Ψ (936)
psi lowercase	Character	ψ		Greek character ψ (968)
psi to Pa	Function (linked ; monadic)	psi→Pa		Convert pounds per square inch to Pascal (Pa:)
Put keycode in buffer	Command	PUTK	PUTK	Copy keycode from register to keyboard buffer for immediate execution (PUTK __ Tam menu)
Put submatrix	Command	PUTM	M.PUT	Put submatrix
pwr to dB	Function (linked ; monadic)	pwr→dB		Convert power ratio to decibel (dB:)
Q	Character	Q		Character Q (81)
q lowercase	Character	q		Character q (113)
Qoppa	Character	Ϡ		Greek character Ϡ (984)
qoppa lowercase	Character	ϡ		Greek character ϡ (985)
QR decomposition	Function (monadic)	M.QR	M.QR	Decompose matrix into unitary (Q) in Y and upper (R) matrix in X (QR factorisation)
qt. to l	Function (linked ; monadic)	qt.→l		Convert quart to liter (l:)
Quadratic means	Command	\bar{x}_{RMS}	\bar{x}_{RMS}	Quadratic means (root mean square) (\bar{x}_{RMS} ; $\bar{x}_{RMS} = (2 \text{ stack levels})$)
Question mark	Character	?		Character ? (63)
Quiet	Flag	QUIET		Beeper is disabled
Quotation mark	Character	"		Character " (34)
R	Character	R		Character R (82)
R acute	Character	Ŕ		International character Ŕ (340)
r acute lowercase	Character	ŕ		International character ŕ (341)
R caron	Character	Ř		International character Ř (344)
r caron lowercase	Character	ř		International character ř (345)
r lowercase	Character	r		Character r (114)
rad to deg	Function (linked ; monadic)	rad→deg		Convert radian to degree (untagged) (deg:)
rad to grad	Function (linked ; monadic)	rad→grad		Convert radian to gradian (untagged) (grad:)
rad/s to RPM	Function (linked ; monadic)	rad/s→RPM		Convert radian per second to rotation per minute (RPM:)
Radix comma	Setting (pgm)	COM,	RCOM, ()	Radix decimal comma
Radix dot	Setting (pgm)	DOT.	RDOT. ()	Radix decimal dot
Radix period	Setting (pgm)	PER.	RPER. (.)	Radix decimal period
Radix wide comma	Setting (pgm)	WCOM,	RWCOM, ()	Radix decimal wide comma
Radix wide dot	Setting (pgm)	WDOT.	RWDOT. ()	Radix decimal wide dot
Radix wide period	Setting (pgm)	WPER.	RWPER. ()	Radix decimal wide period
Raise e to power	Function (monadic)	e^x		Raise e to power
Random integer	Function (dyadic)	RANI#	RANI#	Random integer : lower ENTER upper ; keeping input on stack (X: result, Y: upper, Z: lower)

FullName	Type	Label	Catalog	Description
Random number	Command	RAN#	RAN#	Random number (real)
Range	Setting	RNG	RNG _{6,145}	Range 10 ⁿⁿ (min 99, max 6145) (RNG ____ TamNonReg menu)
Read program	Command	READP	READP	Read program from WRITEP file <program.p47> in FAT (File open dialog (PROGRAMS) ; Inverse : WRITEP)
Real	MENU	REAL	REAL	Functions on real and complex numbers
Real number display format	Variable (longint)	REALDF	REALDF	Real number display format (reserved long integer variable, write protected) (ALL: 0 ; FIX: 1 ; SCI: 2 ; ENG: 3 ; SIG: 4 ; UNIT: 5)
Real one	Command (TAM)	1.		Presented in TAM menus for comparisons (e.g. x< ?) to quickly enter 1.
Real part	Function (monadic)	Re	Re	Real part of complex number
Real R	Character	R		Character R (8477)
Real results	Command	CPXRES0	CPXRES0	Do not allow complex results for real input ; an error will occur for such events (For programming purposes)
Real to complex	Function (dyadic)	RE→CX	RE→CX	Convert reals to complex (in POLAR, using angle tag or ADM)
Real variables	MENU	REALS	REALS	Auto-generated catalog of variables of the specified type: real
Real zero	Command (TAM)	0.		Presented in TAM menus for comparisons (e.g. x< ?) to quickly enter 0.
Real?	Function (monadic)	REAL?	REAL?	Test X is real
Recall (add to timer)	Command (nonpgm)	RCL		Recall register value and add to running stopwatch timer (decimal hours)
Recall (register)	Command	RCL	RCL	Recall value from register or variable can be followed by +, -, ×, ÷ for recall and add, recall and subtract, recall and multiply, recall and divide functions (RCL __ TamStoRcl(Alpha) menu)
Recall and add	Function (monadic)	RCL+	RCL+	Recall register or variable and add X (RCL+ __ TamStoRcl(Alpha) menu)
Recall and divide	Function (monadic)	RCL/	RCL/	Recall register or variable and divide by X (RCL/ __ TamStoRcl(Alpha) menu)
Recall and multiply	Function (monadic)	RCL×	RCL×	Recall register or variable and multiply by X (RCL× __ TamStoRcl(Alpha) menu)
Recall and subtract	Function (monadic)	RCL-	RCL-	Recall register or variable and subtract X (RCL- __ TamStoRcl(Alpha) menu)
Recall configuration	Command	RCLCFG	RCLCFG	Recall configuration from register or variable (RCLCFG __ TamStoRcl(Alpha) menu)
Recall current element	Command	RCLEL	RCLEL	Recall current element
Recall current index	Command	RCLIJ	RCLIJ	Recall current index
Recall maximum	Command	RCL↑	RCL↑	Recall maximum of X and register or variable (RCL↑ __ TamStoRcl(Alpha) menu)
Recall minimum	Command	RCL↓	RCL↓	Recall minimum of X and register or variable (RCL↓ __ TamStoRcl(Alpha) menu)
Recall stack	Command	RCLS	RCLS	Recall complete stack from 4 or 8 registers (RCLS __ TamStoRcl(Alpha) menu)
Recall triple I	Command	RCL 3I	RCL 3I	Copy R96, R97, R98 to X, Y, Z
Recall triple V	Command	RCL 3V	RCL 3V	Copy R93, R94, R95 to X, Y, Z
Recall triple Z	Command	RCL 3Z	RCL 3Z	Copy R90, R91, R92 to X, Y, Z
Reciprocal	Function (monadic)	1/x	1/x	Reciprocal (1/x)
Rectangular	Setting (pgm)	RECT	RECT (•)	Rectangular display of complex numbers (internal value is RECT) (FF X (TAM) ; Info : SBI depends on SBcpX)
Redraw graph	Command	reDraw		Redraw graph for equation
Reduced Planck constant	Constant (#19)	ħ		red.planck ħ = +1,054571817646156391262428003302281 × 10 ⁻³⁴ (Js)
Register A	Shortcut (TAM)	Reg A		TAM shortcut for accessing register A (Shortcut A (C47.21.41))
Register A	Variable (register)	A		Register A (reserved variable)
Register B	Shortcut (TAM)	Reg B		TAM shortcut for accessing register B (Shortcut B (C47.22.41))
Register B	Variable (register)	B		Register B (reserved variable)
Register browser	Browser	REGS	REGS	Browse all registers (+: switch register/variable viewR/S: switch contents/storage viewRCL: recall bottom itemUp/Dn; A..D; I..L; 00..99: navigation)
Register C	Shortcut (TAM)	Reg C		TAM shortcut for accessing register C (Shortcut C (C47.23.41))
Register C	Variable (register)	C		Register C (reserved variable)
Register D	Shortcut (TAM)	Reg D		TAM shortcut for accessing register D (Shortcut D (C47.24.41))
Register D	Variable (register)	D		Register D (reserved variable)
Register I	Shortcut (TAM)	Reg I		TAM shortcut for accessing register I (Shortcut I (C47.33.41))

FullName	Type	Label	Catalog	Description
Register I	Variable (register)	I		Register I (reserved variable)
Register J	Shortcut (TAM)	Reg J		TAM shortcut for accessing register J (Shortcut J (C47.34.41))
Register J	Variable (register)	J		Register J (reserved variable)
Register K	Shortcut (TAM)	Reg K		TAM shortcut for accessing register K (Shortcut K (C47.35.41))
Register K	Variable (register)	K		Register K (reserved variable)
Register L	Shortcut (TAM)	Reg L		TAM shortcut for accessing register L (Shortcut L (C47.36.41))
Register L	Variable (register)	L		Register L (reserved variable)
Register T	Variable (register)	T		Register T (reserved variable)
Register X	Variable (register)	X		Register X (reserved variable)
Register Y	Variable (register)	Y		Register Y (reserved variable)
Register Z	Variable (register)	Z		Register Z (reserved variable)
Regression	MENU	REGR	REGR	Regression functions
Remainder	Function (dyadic)	RMD	RMD	Remainder of division of Y by X
Reset	Command	RESET	RESET	Reset the calculator (Startup using autosaved backup file C47auto.sav)
Reset f/g timers	Command	S.RESET	S.RESET	Safe reset, then toggle ON/OFF all accessibility related options: HOME.3 [*] ; g.2Tp [*] ; SH.4s [*] ; fg.FUL (*)
Reset fitting	Command	ResetF	ResetF	Reset curve fitting
Reset plot	Command	PLTRST	PLTRST	PLSTAT only: reset all plot options and redraw graph (Options include boxes, crosses, lines, scales)
Reset stopwatch	Command	RESET		Reset the stopwatch (⇔)
Return	Command (PEM)	RTN	RTN	Return from (sub)routine to calling routine
Return 2 levels up	Command (PEM)	RTN+1	RTN+1	Return from (sub)routine to 1 level higher than calling routine
Rho	Character	ρ		Greek character ρ (929)
rho lowercase	Character	ρ		Greek character ρ (961)
Right	Symbol	→		Move right (navigation) or arrow character (alpha selection menus) (8594)
Right	Arrow	→		Move right (navigation) or arrow character (alpha selection menus)
Right angle	Character	∟		Character ∟ (8735)
Right bit mask	Function (monadic)	MASKR	MASKR	Set right n bits to use as mask (MASKR __ TamNonReg menu)
Right curly bracket	Character	}		Character } (125)
Right double quotation mark	Character	”		Character ” (8221)
Right French quotation mark	Character	»		Character » (187)
Right justify	Function (monadic)	RJ	RJ	Right justify (within word size) ; returns shift in X and result in Y
Right parenthesis	Symbol)		Close (right) parenthesis (41)
Right single quotation mark	Character	'		Character ' (8217)
Right square bracket	Character]		Character] (93)
Right-right	Arrow	→		Move all the way to the right (navigation) or arrow character (alpha selection menus)
Roll down	Command	R↓	R↓	Roll down stack
Roll up	Command	R↑	R↑	Roll up stack
Root fit	Setting	RootF	RootF []	Root curve fitting
Rotate left	Function (monadic)	RL	RL	Rotate left with number of bits (trailing input) (RL __ TamNonReg menu)
Rotate left (1)	Function (monadic)	RL1	RL1	Shortcut to rotate left (1 bit)
Rotate left through Carry	Function (monadic)	RLC	RLC	Rotate left through Carry (RLC __ TamNonReg menu)
Rotate right	Function (monadic)	RR	RR	Rotate right with number of bits (trailing input) (RR __ TamNonReg menu)
Rotate right (1)	Function (monadic)	RR1	RR1	Shortcut to rotate right (1 bit)
Rotate right through Carry	Function (monadic)	RRC	RRC	Rotate right through Carry (RRC __ TamNonReg menu)
Round	Function (monadic)	ROUND	ROUND	Rounds to current display format (type real)
Round to decimal places	Function (monadic)	RDP	RDP	Rounds to n decimal places (RDP __ TamNonReg menu)
Round to integer	Function (monadic)	ROUNDI	ROUNDI	Rounds to next integer (max 1000 digits)
Round to significant digits	Function (monadic)	RSD	RSD	Rounds to number of significant digits, subject to rounding mode (RMODE) (RSD __ TamNonReg menu)

FullName	Type	Label	Catalog	Description
Rounding mode	Setting (pgm)	RMODE	RMODE ₀	Set floating point rounding mode, used only for RSD ; also used when converting from the extended precision internal format to packed reals (RMODE _ TamNonReg menu)
Rounding mode	Command	RMODE?	RMODE?	Floating point rounding mode ; set by RMODE
Row norm	Command	RNORM	RNORM	Row norm
Row sum	Command	RSUM	RSUM	Row sum
RPM to deg/s	Function (linked ; monadic)	RPM→deg/s		Convert rotation per minute to degree per second (deg/s:)
RPM to rad/s	Function (linked ; monadic)	RPM→rad/s		Convert rotation per minute to radian per second (rad/s:)
Run/Stop	Command	R/S		Run/Stop (Program)
Run/Stop	Command	STOP	STOP	Run/Stop (Program)
Rydberg constant	Constant (#42)	R _∞		c.rydberg R _∞ = +1.097373156816 × 10 ⁷ (/mol)
S	Character	Š		Character Š (83)
S acute	Character	Ṧ		International character Ṧ (346)
s acute lowercase	Character	ṧ		International character ṧ (347)
S caron	Character	Š̂		International character Š̂ (352)
s caron lowercase	Character	š̂		International character š̂ (353)
S cedilla	Character	Š̃		International character Š̃ (350)
s cedilla lowercase	Character	š̃		International character š̃ (351)
s lowercase	Character	s		Character s (115)
s to ftn	Function (linked ; monadic)	s→ftn		Convert second to fortnight (ftn:)
s to year	Function (linked ; monadic)	s→year		Convert second to year (year:)
S _{xy}	Command	S _{xy}	S _{xy}	Sample covariance
Sampi	Character	Ϻ		Greek character Ϻ (992)
sampi lowercase	Character	ϻ		Greek character ϻ (993)
Sample standard deviation	Command	s	s	Sample standard deviation (s _x ; s _y = (2 stack levels))
Save full backup	Command (nonpgm)	SAVE	SAVE	Save full backup to file C47.sav in FAT (SAVFILES ; Inverse : LOAD)
Save program	Command	X.SAVE		Save XEQC program (X.SAVE __ TamNonReg menu)
Save state file	Command (nonpgm)	SAVEST	SAVEST	Save state file to file <state.s47> in FAT (File save dialog (STATE) ; Inverse : LOADST)
Scatter plot	MENU (item)	SCATR	SCATR	Scatter plot of measurments
Scatter plot	MENU	SCATR		Scatter plot of measurments (CENTRL (re)starts menu SCATR)
Scattering factor for a lognormal population	Command	ε _p	ε _p	Scattering factor for a lognormal population (ε _{px} ; ε _{py} = (2 stack levels))
Scattering factor for a lognormal sample	Command	ε	ε	Scattering factor for a lognormal sample (ε _x ; ε _y = (2 stack levels))
Scattering factor of the geometric mean	Command	ε _m	ε _m	Scattering factor of the geometric mean (ε _{mx} ; ε _{my} = (2 stack levels))
Scientific display large reals	Setting (pgm)	SCIOVR	SCIOVR (•)	Change display to SCI for reals too large to display in full (FF A)
Scientific notation	Setting (pgm)	SCI	SCI ()	Set numeric display mode to SScientific notation with nn+1 digits (SCI __ TamNonReg menu)
Screenshot	Command	SNAP	SNAP	Save screenshot as image to bitmap file in FAT ; if executed from the NORMAL keyboard (C47.81.13) saves contents of stack or alpha buffer as text to data file in FLASH memory ; plays clicking sound ([f/G] + [E])
Script	Command (cyclic)	SCR		Cycle normal script - superscript - subscript (locking)
Scroll down/Single step	Command (nonpgm)	≡▼		Single Step
Scroll up/Backstep	Command (nonpgm)	≡▲		Back Step
Second derivative	MENU	f''	f''	Second derivative
Seconds	Function (monadic)	SEC	SEC	Seconds (of time)
Section	Character	§		Character § (167)
Seed	Command	SEED	SEED	Set random seed (0..1] ; for values less than or equal to 0, the seed is derived from the internal clock
Select FP separator	MENU	FPART	FPART	Select fractional part separator

FullName	Type	Label	Catalog	Description
Select FP separator	MENU (item)	FPART _L		Select fractional part separator ; showing current separator ; symbol ø for no separator (FPART)
Select IP separator	MENU	IPART	IPART	Select integer part separator
Select IP separator	MENU (item)	IPART _L		Select integer part separator ; showing current separator ; symbol ø for no separator (IPART)
Select radix	MENU (item)	RADIX.		Select radix ; showing current radix (RADIX)
Select radix	MENU	RADIX	RADIX	Select radix
Semi-major axis of the Earth	Constant (#45)	S _a		majax.earth S _a = +6.378137 × 10 ⁶ (m)
Semi-minor axis of the Earth	Constant (#46)	S _b		minax.earth S _b = +6.3567523142 × 10 ⁶ (m)
Semicolon	Character	;		Character ; (59)
Serial I/O	Character	↕		Character ↕ (8597)
Serial I/O	Flag (system)	RUNIO		Serial input/output active (SBI depends on SBser)
Set ADM to D.MS	Setting (legacy)	D.MS	D.MS (.)	Set ADM to sexagesimal degrees mode (SBI depends on SBang)
Set ADM to DEG	Setting (pgm)	DEG	DEG (°)	Set ADM to degrees mode (SBI depends on SBang)
Set ADM to GRAD	Setting (pgm)	GRAD	GRAD (°)	Set ADM to gradians mode (SBI depends on SBang)
Set ADM to MULπ	Setting (legacy)	MULπ	MULπ (π)	Set ADM to multiple of π radians mode (SBI depends on SBang)
Set ADM to RAD	Setting (pgm)	RAD	RAD (°)	Set ADM to radians mode (SBI depends on SBang)
Set all (models)	Command	ALLF	ALLF	Pre-selects all models (opposite from ResetF)
Set bit	Function (monadic)	SB	SB	Set bit n (SB __ TamNonReg menu)
Set current index	Command	STOIJ	STOIJ	Set current index
Set D.MS tag or convert to D.MS	Function (monadic)	⇒D.MS	⇒D.MS	If untagged, set tag to D.MS ; if tagged, convert X to D.MS ; does not change ADM ; X considered as dd.mmss (° ' ")
Set date	Setting (pgm)	SETDAT	SETDAT	Set date
Set DEG tag or convert to DEG	Function (monadic)	⇒DEG	⇒DEG	If untagged, set tag to DEG ; if tagged, convert X to degrees ; does not change ADM (°)
Set flag	Command	SF	SF	Set flag (SF __ TamFlag menu)
Set GRAD tag or convert to GRAD	Function (monadic)	⇒GRAD	⇒GRAD	If untagged, set tag to GRAD ; if tagged, convert X to GRAD ; does not change ADM (°)
Set Julian-Gregorian (transition)	Function (monadic)	J/G	J/G	Set the day that Julian date changes over to Gregorian date (using date in X) (Input real number according to function x→DATE)
Set MULπ tag or convert to MULπ	Function (monadic)	⇒MULπ	⇒MULπ	If untagged, set tag to MULπ ; if tagged, convert X to MULπ ; does not change ADM (π)
Set numeric (one digit)	Command	(Digit)		Set numeric (one digit) (gShiftedAIM [f/g])
Set RAD tag or convert to RAD	Function (monadic)	⇒RAD	⇒RAD	If untagged, set tag to RAD ; if tagged, convert X to RAD ; does not change ADM (°)
Set significant digits	Setting (pgm)	SDIGS	SDIGS ₃₄	Set the number of significant digits (1 ... 34) for rounding after each operation ; sets tolerance of Solver and CONVg? ; value of 0 sets maximum precision (34) (SDIGS __ TamNonReg menu)
Set time	Setting (pgm)	SETTIM	SETTIM	Set time
Settings to text file	Command (strike)	set>TXT	set>TXT	Settings to text file
Setup	MENU	SETUP	SETUP	System (mode) settings with status indication and modification
Shade curve	Setting	[AREA		Shade the area underneath the graphical integral curve
Shift Digits Left	Function (monadic)	SDL	SDL	Shift digits to the left (SDL __ TamNonReg menu)
Shift Digits Right	Function (monadic)	SDR	SDR	Shift digits to the right (SDR __ TamNonReg menu)
Shift f	f-shift	[f]		Shift f (yellow) (Placement depends on SBshfR)
Shift f/g	fg-shift	[f/g]		Single press: shift f (yellow) ; double press: shift g (blue)
Shift g	g-shift	[g]		Shift g (blue) (Placement depends on SBshfR)
Shift left	Function (monadic)	SL	SL	Shift bits right (SL __ TamNonReg menu)
Shift left (1)	Function (monadic)	SL1	SL1	Shortcut to shift left (1 bit)
Shift right	Function (monadic)	SR	SR	Shift bits right (SR __ TamNonReg menu)
Shift right (1)	Function (monadic)	SR1	SR1	Shortcut to shift right (1 bit)
Shift time-out 4s	Setting	SH.4s	SH.4s [.]	Set shift to time out after 4 seconds
short cwt to kg	Function (linked ; monadic)	short cwt→kg		Convert short hundredweight to kilogram (kg:)
short ton to kg	Function (linked ; monadic)	short ton→kg		Convert short ton to kilogram (kg:)

FullName	Type	Label	Catalog	Description
Shortint variables	MENU	S.INTS	S.INTS	Auto-generated catalog of variables of the specified type: short integer
Show	Command	SHOW	SHOW	Show item in maximum detail, favouring register data type (tag)
Show all prefixes	Setting	PFX.All	PFX.All [•]	Show all SI unit prefixes in UNIT (prefix) display mode: 10^{-30} to 10^{30} ; if OFF, the range is 10^{-15} to 10^{15}
Show angular mode symbol	Flag	SBang		Display symbol "°" in status bar
Show battery capacity	Flag	SBbatV		Display battery voltage with battery length proportional to 2.054 V to 3.045 V (replacing the standard LOWBAT icon)
Show complex mode	Flag	SBcpx		Display status of RECT : Rectangular ; POLAR : Polar in status bar
Show complex result	Flag	SBcr		Display status of CPXRES : Complex results in status bar
Show date	Flag	SBdate		Display date in status bar
Show fraction and base mode	Flag	SBfrac		Display status of FRACT : Fraction ; # : Number (base) in status bar
Show integer mode	Flag	SBint		Display status of WSIZE : Word size and complement/unsigned in status bar
Show matrix mode	Flag	SBmx		Display status of WRAP : Wrap (matrix edit) ; GROW : Grow (matrix edit) in status bar
Show oc mode	Flag	SBoc		Display status of OVERFL : Overflow ; CARRY : Carry in status bar
Show printer	Flag	SBprn		Display status of PRINTS : Printing in status bar
Show serial io	Flag	SBser		Display status of RUNIO : Serial I/O in status bar
Show shift on the right	Flag	SBshfR		Display f ; g on the right side of the status bar
Show stack size	Flag	SBss		Display status of SSIZE8 : Stack Size 8 in status bar
Show stopwatch	Flag	SBclk		Display status of RUNTIM : Stopwatch running in status bar
Show time	Flag	SBtime		Display time in status bar
Show tvn mode	Flag	SBtvm		Display status of End : TVM end payments ; Begin : TVM begin payments in status bar
Shuffle stack	Command	↯	↯	Shuffle stack: replace X Y Z T by selection of X Y Z T (↯ ____ TamShuffle menu)
Sigma	Character	Σ		Greek character Σ (931)
sigma lowercase	Character	σ		Greek character σ (963)
Sigma-	Command	Σ-	Σ-	Remove data from the statistics matrix (STATS) (nnn data points)
Sigma+	Command	Σ+	Σ+	Enter data into the statistics matrix (STATS) (nnn data points)
Sign	Function (monadic)	sign	sign	Sign is -1 for negative numbers, 0 for zero, +1 for positive numbers
Sign and mantissa	Setting (pgm)	SIGNMT	SIGNMT ()	Set sign and mantissa mode for shortint (SBI depends on SBint)
Signed 16 bits	Setting (pgm)	S16		Shortcut to set word size to 16 bits signed
Signed 32 bits	Setting (pgm)	S32		Shortcut to set word size to 32 bits signed
Signed 6 bits	Setting (pgm)	S06		Shortcut to set word size to 6 bits signed
Signed 64 bits	Setting (pgm)	S64		Shortcut to set word size to 64 bits signed
Signed 8 bits	Setting (pgm)	S08		Shortcut to set word size to 8 bits signed
Significant Digits	Command	SDIGS?	SDIGS?	Number of significant digits ; set by SDIGS
Significant digits notation	Setting (pgm)	SIG	SIG ()	Set numeric display mode to SIGNificant notation with nn+1 digits ; switching over to scientific or engineering notation when number of rounded trailing zeros exceeds group size (IPGRP) (SIG __ TamNonReg menu)
Sinc	Function (monadic)	sinc	sinc	(Sine of X) / X
Sinc pi	Function (monadic)	sincπ	sincπ	(Sine of π * X) / (π * X)
Sine	Function (monadic)	SIN	SIN	Sine
Sine	Function (monadic)	SIN	SIN	Sine
Skip	Command (PEM)	SKIP	SKIP	Skip n program steps (SKIP __ TamNonReg menu)
Slash	Character	/		Character / (47)
Slow mode	Flag	SLOW		Slow mode active to limit battery drain
SLVQ	Command	SLVQ	SLVQ	Solves the quadratic equation (parameters X = a, Y = b, Z = c)
Solve	Command	SOLVE	SOLVE	Solve the equation ; use X, Y as initial guesses ; fill all stack registers with X (SOLVE __ TamLbl(Alpha) menu)
Solver	MENU	Solver	Solver	Solver, with iteration counter, interrupt by keypress ; tolerance set by SDIGS (Special use of registers R81-R98, see Ref: Registers)
Solving	Flag (system)	SOLVING		Solver is running

FullName	Type	Label	Catalog	Description
Sort registers	Command	R-SORT	R-SORT	Sort registers (sss.nn means sort registers from sss through sss + nn - 1)
Space	Character	_		Character _ (9251)
Special key assignments	MENU	Σ+NRM	I	Submenu for special key assignments to Sigma+ key in normal mode
Special results	Setting	SPCRES	SPCRES [.]	Set to allow special results of calculations (infinity, not-a-number) ; an error will not occur for such events (FF D)
Special results	Flag	SPCRES		Set to allow special results of calculations (infinity, not-a-number) ; an error will not occur for such events (FF D)
Special results	Command	SPCRES1	SPCRES1	Allow special results of calculations (infinity, not-a-number) ; an error will not occur for such events (For programming purposes)
Special?	Function (monadic)	SPEC?	SPEC?	Test X is special ($\pm\infty$ or NaN)
Speed conversion	MENU	Speed:	Speed:	Convert between units of speed
Speed of light (vacuum)	Constant (#4)	c		lightspeed c = $+2.99792458 \times 10^8$ (m/s)
Square	Function (monadic)	x^2	x^2	Square of X
Square matrix?	Function (monadic)	M.SQR?	M.SQR?	Test matrix is square
Square root	Symbol	√		Square root (8730)
Square root	Function (legacy, monadic)	SQRT	SQRT	Square root (8730)
Square root	Function (monadic)	√	√	Square root (8730)
Square root	Function (monadic)	\sqrt{x}	\sqrt{x}	Square root
Stack	MENU	STK	STK	Stack functions
Stack size	Command	SSIZE?	SSIZE?	Number of stack registers currently allocated (4 or 8) ; set by SSIZE4 ; SSIZE8
Stack size 4	Setting (pgm)	SSIZE4	SSIZE4 ()	Set stack size to 4 registers (SBI depends on SBss)
Stack Size 8	Setting (pgm)	SSIZE8	SSIZE8 (•)	Set stack size to 8 registers (SBI depends on SBss)
Stack Size 8	Flag	SSIZE8		Set for 8 stack registers, clear for 4 registers (SBI depends on SBss)
Stack to date	Function (triadic)	→DATE	→DATE	Convert stack values X, Y, Z to date according to DISP or CLK settings for date format (weekday)
Stack to Time	Function (monadic)	→TIME	→TIME	Convert to time
Standard atmospheric pressure	Constant (#37)	p_0		press.atm $p_0 = +1.01325 \times 10^5$ (Pa)
Standard Earth acceleration	Constant (#17)	g_0		acc.earth $g_0 = +9.80665$ (m/s ²)
Standard error of the mean	Command	s_m	s_m	Standard error of the mean (s_{mx} ; $s_{my} = (2 \text{ stack levels})$)
Standard error of the weighted mean	Command	s_{mw}	s_{mw}	Standard error of the weighted mean ($s_{mw} =$)
Standard errors	Command	s(a)	s(a)	Standard errors of line fitted ; s(a_0) in X ; s(a_1) in Y (Works for EXPF, LINF, LOGF, ORTOHOF, POWERF ; does not work for CAUCHF, GAUSSF, HYPF, PARABF, ROOTF)
Standard normal (inverse)	Function (tbd)	Φ^{-1}	Φ^{-1}	Standard probability inverse function
Standard normal (probability)	MENU	Φ :	Φ :	Standard normal probability distribution ($\mu = \text{mean} = 0$; $\sigma = \text{standard deviation} = 1$)
Standard normal cdf (lower)	Function (tbd)	Φ_L	Φ_L	Standard cumulative distribution (lower tail)
Standard normal cdf (upper)	Function (tbd)	Φ_U	Φ_U	Standard cumulative distribution (upper tail)
Standard normal pdf	Function (tbd)	φ_p	φ_p	Standard probability density function
Standard temperature	Constant (#50)	T_0		temp.stand $T_0 = +2.7315 \times 10^2$ (K)
Star (Wye) to Delta	Function (triadic)	$Y \rightarrow \Delta$	$Y \rightarrow \Delta$	Convert delta connected impedances X, Y, Z to star impedances X, Y, Z ($\Delta \rightarrow Y$)
Statistics	MENU	STAT	STAT	Statistics functions
Statistics matrix	Variable (matrix)	STATS	STATS	Reserved matrix variable for statistics matrix (STATS) (Enter data using function Σ+)
Stats range	Command	x_{RANGE}	x_{RANGE}	Range for both X and Y ; this is equal to MAX - MIN of the statistics matrix (STATS) (rg_x ; $rg_y = (2 \text{ stack levels})$)
Status	Command	STATUS	STATUS	Show status pages (memory, flags and settings) (Flags)
Stefan-Boltzmann constant	Constant (#72)	σ_B		c.stephol $\sigma_B = +5,670374419184429453970996731889231 \times 10^{-8}$ (W/m ² K ⁴)
STO/RCL configuration	Command (TAM)	Config		Presented in TAM menus when STO/RCL is active to quickly select STOCFG/RCLCFG (STO/RCL Configuration) (STOCFG/RCLCFG __ TamStoRcl(Alpha) menu)
STO/RCL maximum	Command (TAM)	Max		Presented when TamStoRcl(Alpha) menu is active to quickly select STO/RCL (maximum of X and source) (STO/RCL __ TamStoRcl(Alpha) menu)

FullName	Type	Label	Catalog	Description
STO/RCL minimum	Command (TAM)	Min		Presented when TamStoRcl(Alpha) menu is active to quickly select STO \downarrow /RCL \downarrow (minimum of X and source) (STO \downarrow /RCL \downarrow __ TamStoRcl(Alpha) menu)
STO/RCL stack	Command (TAM)	Stack		Presented in TAM menus when STO/RCL is active to quickly select STOS/RCLS (STO/RCL Stack) (STOS/RCLS __ TamStoRcl(Alpha))
stone to kg	Function (linked ; monadic)	stone \rightarrow kg		Convert stone to kilogram (kg):
Stopwatch	App (item)	STOPW	STOPW	Stopwatch
Stopwatch	Character	⌚		Character ⌚ (9201)
Stopwatch	App	STOPW		Stopwatch (Running time, counter ; start/stop using R/S)
Stopwatch running	Flag (system)	RUNTIM		Set if the stopwatch is running (SBI not implemented yet ; Info : SBI depends on SBclk)
Store (register)	Command	STO	STO	Store value in register or variable ; can be followed by +, -, \times , \div for add into, subtract into, multiply into, divide into functions (STO __ TamStoRcl(Alpha) menu)
Store configuration	Command	STOCFG	STOCFG	Store configuration in register or variable (STOCFG __ TamStoRcl(Alpha) menu)
Store maximum	Command	STO \uparrow	STO \uparrow	Store maximum of X and register or variable (STO \uparrow __ TamStoRcl(Alpha) menu)
Store minimum	Command	STO \downarrow	STO \downarrow	Store minimum of X and register or variable (STO \downarrow __ TamStoRcl(Alpha) menu)
Store stack	Command	STOS	STOS	Store entire stack in 4 or 8 registers (STOS __ TamStoRcl(Alpha) menu)
Store triple I	Command	STO 3I	STO 3I	Copy X, Y, Z to R96, R97, R98
Store triple V	Command	STO 3V	STO 3V	Copy X, Y, Z to R93, R94, R95
Store triple Z	Command	STO 3Z	STO 3Z	Copy X, Y, Z to R90, R91, R92
Store X into current element	Command	STOEL	STOEL	Store X into current element
String variables	MENU	STRING	STRING	Auto-generated catalog of variables of the specified type: string
String?	Function (monadic)	STRI?	STRI?	Test X is text string
Student's t (inverse)	Function (tbd)	$t^{-1}(p)$	$t^{-1}(p)$	Student's t probability inverse function
Student's t cdf (lower)	Function (tbd)	$t_{\downarrow}(x)$	$t_{\downarrow}(x)$	Student's t cumulative distribution (lower tail)
Student's t cdf (upper)	Function (tbd)	$t_{\uparrow}(x)$	$t_{\uparrow}(x)$	Student's t cumulative distribution (upper tail)
Student's t distribution	MENU	t:	t:	Student's t probability distribution (RegI = v = degrees of freedom)
Student's t pdf	Function (tbd)	$t_p(x)$	$t_p(x)$	Student's t probability density function
Subscript	Command	Sub		Subscript
Subscript E outline	Character	Ɛ		Character Ɛ (8307)
Subscript Earth	Character	⊕		Character ⊕ (8853)
Subscript infinity	Character	∞		Character ∞ (8351)
Subscript SUN	Character	☉		Character ☉ (8858)
Subtract	Function (dyadic)	[-]	-	Subtract X from Y
Subtract into	Function (monadic)	STO-	STO-	Subtract X from register or variable (STO- __ TamStoRcl(Alpha) menu)
Sum (programmable)	Command	Σ_n	Σ_n	Real or complex sum using specified program, with iteration counter, interrupt by keypress (Σ_n __ TamLbl(Alpha) menu ; <from> ENTER <to> ENTER <step>)
Summation of \ln^2x	Command	$\Sigma \ln^2x$	$\Sigma \ln^2x$	Summation of \ln^2x using statistics matrix (STATS)
Summation of \ln^2y	Command	$\Sigma \ln^2y$	$\Sigma \ln^2y$	Summation of \ln^2y using statistics matrix (STATS)
Summation of $\ln x$	Command	$\Sigma \ln x$	$\Sigma \ln x$	Summation of $\ln x$ using statistics matrix (STATS)
Summation of $\ln x \cdot \ln y$	Command	$\Sigma \ln x \cdot \ln y$	$\Sigma \ln x \cdot \ln y$	Summation of $\ln x \cdot \ln y$ using statistics matrix (STATS)
Summation of $\ln y$	Command	$\Sigma \ln y$	$\Sigma \ln y$	Summation of $\ln y$ using statistics matrix (STATS)
Summation of x	Command	Σx	Σx	Summation of x using statistics matrix (STATS)
Summation of $x \cdot \ln y$	Command	$\Sigma x \cdot \ln y$	$\Sigma x \cdot \ln y$	Summation of $x \cdot \ln y$ using statistics matrix (STATS)
Summation of x^2	Command	Σx^2	Σx^2	Summation of x^2 using statistics matrix (STATS)
Summation of $x^2 \cdot \ln y$	Command	$\Sigma x^2 \cdot \ln y$	$\Sigma x^2 \cdot \ln y$	Summation of $x^2 \cdot \ln y$ using statistics matrix (STATS)
Summation of x^2y	Command	Σx^2y	Σx^2y	Summation of x^2y using statistics matrix (STATS)
Summation of x^2y^{-1}	Command	Σx^2y^{-1}	Σx^2y^{-1}	Summation of x^2y^{-1} using statistics matrix (STATS)
Summation of x^3	Command	Σx^3	Σx^3	Summation of x^3 using statistics matrix (STATS)
Summation of x^4	Command	Σx^4	Σx^4	Summation of x^4 using statistics matrix (STATS)
Summation of x^{-2}	Command	Σx^{-2}	Σx^{-2}	Summation of x^{-2} using statistics matrix (STATS)
Summation of xy	Command	Σxy	Σxy	Summation of xy using statistics matrix (STATS)

FullName	Type	Label	Catalog	Description
Summation of xy^{-1}	Command	Σxy^{-1}	Σxy^{-1}	Summation of xy^{-1} using statistics matrix (STATS)
Summation of x^{-1}	Command	Σx^{-1}	Σx^{-1}	Summation of x^{-1} using statistics matrix (STATS)
Summation of $x^{-1} \cdot \ln y$	Command	$\Sigma x^{-1} \cdot \ln y$	$\Sigma x^{-1} \cdot \ln y$	Summation of $x^{-1} \cdot \ln y$ using statistics matrix (STATS)
Summation of y	Command	Σy	Σy	Summation of y using statistics matrix (STATS)
Summation of $y \cdot \ln x$	Command	$\Sigma y \cdot \ln x$	$\Sigma y \cdot \ln x$	Summation of $y \cdot \ln x$ using statistics matrix (STATS)
Summation of y^2	Command	Σy^2	Σy^2	Summation of y^2 using statistics matrix (STATS)
Summation of y^{-2}	Command	Σy^{-2}	Σy^{-2}	Summation of y^{-2} using statistics matrix (STATS)
Summation of y^{-1}	Command	Σy^{-1}	Σy^{-1}	Summation of y^{-1} using statistics matrix (STATS)
Superscript	Command	Super		Superscript
Superscript asterisk	Character	*		Character * (8335)
Superscript infinity	Character	∞		Character ∞ (8350)
surveyft.us to m	Function (linked ; monadic)	surveyft.us→m		Convert US survey foot to meter (m)
Swap registers	Command	R-SWAP	R-SWAP	Swap registers (sss.nddd means swap registers from sss through sss + nn - 1 with registers ddd through ddd + nn - 1)
Swap rows	Command	R↔R	M.R↔R	Swap rows
Swap T	Command	t↔	t↔	Swap T and register (t↔ __ Tam menu)
Swap X	Command	x↔	x↔	Swap X and register (x↔ __ Tam menu)
Swap X	Command	X.SWAP	X.SWAP	Swap contents of X register with input of alpha or equation editor
Swap X and Y	Command	x↔y	x↔y	Swap register X and register Y
Swap Y	Command	y↔	y↔	Swap Y and register (y↔ __ Tam menu)
Swap Z	Command	z↔	z↔	Swap Z and register (z↔ __ Tam menu)
Switch on	Command	ON		Switch on calculator (Switched OFF + [EXIT])
System (exit)	Command	SYSTEM	SYSTEM	Exit calculator (causing reset) and enter DMCP (hardware only) ; confirmation dialog "Are you sure?" [Y/N] (Autosaves backup file C47auto.sav to FAT)
System flags	MENU (TAM ; ASM)	SYS.FL	SYS.FL	Presented in TAM menus for commands accessing system flags (Type characters 1-2 to search ; If SYS.FL is accessed from CAT.MENUS, soft buttons return flag status, otherwise toggle flag status)
sz lowercase	Character	ß		International character ß (223)
sz lowercase	Character	ß		International character ß (223)
T	Character	T		Character T (84)
t apostrophe lowercase	Character	t̂		International character t̂ (357)
T caron	Character	ť		International character ť (356)
T cedilla	Character	ṭ		International character ṭ (354)
t cedilla lowercase	Character	ṭ		International character ṭ (355)
t lowercase	Character	t		Character t (116)
T-register	Command (TAM)	T		Presented in TAM menus for commands accessing stack level T
TAM mode	MENU (TAM)	Tam		Transient alpha mode is activated for trailing input (general)
TAM mode (alpha)	MENU (TAM)	TamAlpha		Transient alpha mode is activated for trailing input (general + alpha)
TAM mode CMP	MENU (TAM)	TamCmp		Transient alpha mode is activated for trailing input (compare)
TAM mode CMP (alpha)	MENU (TAM)	TamCmpAlpha		Transient alpha mode is activated for trailing input (compare + alpha)
TAM mode FLAG	MENU (TAM)	TamFlag		Transient alpha mode is activated for trailing input (flag)
TAM mode INDIRECT	MENU (TAM)	TamNonRegInd		Transient alpha mode is activated for trailing input (indirect) (Activated from TAM menus by Indirection (→))
TAM mode KEY	MENU (TAM)	TamKey		Transient alpha mode is activated for trailing input (KEYG, KEYX)
TAM mode LABEL	MENU (TAM)	TamLabel		Transient alpha mode is activated for trailing input (LBL/GT0/XEQ) (AIM (α) activates TamLblAlpha)
TAM mode LABEL (alpha)	MENU (TAM)	TamLblAlpha		Transient alpha mode is activated for trailing input (LBL/GT0/XEQ + alpha) (Activated from TamLabel by AIM (α))
TAM mode SHUFFLE	MENU (TAM)	TamShuffle		Transient alpha mode is activated for trailing input (shuffle)

FullName	Type	Label	Catalog	Description
TAM mode STORCL	MENU (TAM)	TamStoRcl		Transient alpha mode is activated for trailing input (ST0, RCL) (AIM (α) activates TamStoRclAlpha)
TAM mode STORCL (alpha)	MENU (TAM)	TamStoRclAlpha		Transient alpha mode is activated for trailing input (ST0, RCL + alpha) (Activated from TamStoRcl by AIM (α))
TAM mode VALUE	MENU (TAM)	TamNonReg		Transient alpha mode is activated for trailing input (value) (Indirection (→) activates TamNonRegInd)
Tangent	Function (monadic)	TAN	TAN	Tangent
Tangent	Function (monadic)	TAN		Tangent
Tau	Character	τ		Greek character τ (932)
tau lowercase	Character	τ		Greek character τ (964)
Tenths of seconds	Setting	0.1s		Show running time with tenths of seconds
Tera	Command (nonpgm)	•T		Factor 10 ¹²
Testing	MENU	TEST	TEST	Testing functions
Theta	Character	θ		Greek character θ (920)
theta lowercase	Character	θ		Greek character θ (952)
Ticks	Command	TICKS	TICKS	Number of ticks counted since calculator was turned on (one tick is 10 ms)
Tilde	Character	~		Character ~ (126)
Time	Command	TIME	TIME	Current time
Time display 24h	Flag	TDM24		Set for 24h time display, clear for 12h time
Time display format	Setting	TDISP	TDISP ₀	Set time display format: 0 = full ; 1,2 = hours/minutes ; 3 = hours/minutes/seconds ; 4,5,6 = 1,2,3 decimal places for the seconds (TDISP _ TamNonReg menu)
Time to stack	Function (monadic)	TIME→	TIME→	Convert time to hours (24h), minutes, seconds in stack
Time value of money	MENU	TVM	TVM	Time value of money functions
Time variables	MENU	TIMES	TIMES	Auto-generated catalog of variables of the specified type: time
To integer	Function (cyclic ; nonpgm ; m	→I		Convert to long integer/short integer (cyclic, max 1000 digits) (Shortint indicated by subscript ₁₀ ; can show TI: Overfl>nbits (n=1..64))
To polar	Function (monadic ; dyadic)	→P	→POLAR	Transform rectangular to polar coordinates (stack conventions according to flag RP _{HP} ; transform complex number to polar notation (monadic) and set POLAR tag (r = ; θ = (2 stack levels))
To rectangular	Function (monadic ; dyadic)	→R	→RECT	Transform polar to rectangular coordinates (stack conventions according to flag RP _{HP} ; transform complex number to rectangular notation (monadic) and set RECT tag (x : Re = ; y : Im = (2 stack levels))
Toggle numlock	Command	TogNum		Toggle numlock (Triple AIM [f/g])
ton to kg	Function (linked ; monadic)	ton→kg		Convert ton to kilogram (kg)
Tone	Command	TONE	TONE	Tone (0-11) (TONE _ TamNonReg menu)
Top?	Command	TOP?	TOP?	Test program pointer is in top routine (as opposed to in subroutine)
torr to Pa	Function (linked ; monadic)	torr→Pa		Convert torr to Pascal (Pa)
Total periods	Variable (real)	N _{PER}	N _{PER}	Total number of payment or compounding periods for loan or investment
tr.oz to g	Function (linked ; monadic)	tr.oz→g		Convert troy ounce to gram (g)
Tracing	Flag	TRACE		Printing in trace mode (FF T (TAM))
Transpose matrix	Function (monadic)	[M] ^T	[M] ^T	Transpose matrix
Transpose matrix	Function (legacy)	TRANS	TRANS	Transpose matrix
Transposed	Character	τ		Character τ (8868)
Trigonometry	MENU	TRG	TRG ₄₇	Trigonometry functions (other layouts)
Trigonometry	MENU (47)	TRG	TRG ₄₇	Trigonometry and hyperbolic functions
Trigonometry	MENU	TRIG	TRIG	Trigonometry and hyperbolic functions
Triple I = V / Z	Command	V÷Z	3V÷3Z	X = R93 / R90 ; Y = R94 / R91 ; Z = R95 / R92
Triple V = I x Z	Command	I×Z	3I×3Z	X = R96 × R90 ; Y = R97 × R91 ; Z = R98 × R92
Triple Z = V / I	Command	V÷I	3V÷3I	X = R93 / R96 ; Y = R94 / R97 ; Z = R95 / R98

FullName	Type	Label	Catalog	Description
TVM begin payments	Setting (pgm)	Begin	BeginP ()	Payments at the beginning of each period (time value of money) (SBI depends on SBTvm)
TVM end payments	Setting (pgm)	End	ENDP (•)	Payments at the end of each period (time value of money) (SBI depends on SBTvm)
TVM end payments	Flag (system)	ENDPMT		Set for payments at the end of each period, clear for beginning (time value of money) (SBI depends on SBTvm)
U	Character	U		Character U (85)
U acute	Character	Ú		International character Ú (218)
u acute lowercase	Character	ú		International character ú (250)
U breve	Character	Û		International character Û (364)
u breve lowercase	Character	û		International character û (365)
U circumflex	Character	Û		International character Û (219)
u circumflex lowercase	Character	û		International character û (251)
U diaeresis	Character	Ü		International character Ü (220)
u diaeresis lowercase	Character	ü		International character ü (252)
U grave	Character	Ù		International character Ù (217)
u grave lowercase	Character	ù		International character ù (249)
u lowercase	Character	u		Character u (117)
U macron	Character	Ū		International character Ū (362)
u macron lowercase	Character	ū		International character ū (363)
U ogonek	Character	Ų		International character Ų (370)
u ogonek lowercase	Character	ų		International character ų (371)
U ring	Character	Ů		International character Ů (366)
u ring lowercase	Character	ů		International character ů (367)
U tilde	Character	Ũ		International character Ũ (360)
u tilde lowercase	Character	ũ		International character ù (361)
UK formatting	Setting (pgm)	UK	SETUK	Set to UK regional formats (date, time, calendar, number formatting) (First Gregorian day set: 14.09.1752)
Underscore	Character	_		Character _ (95)
Undo	Command	↶	UNDO	Restore complete stack and LASTx register
Unit (prefix) notation	Setting (pgm)	UNIT	UNIT ()	Set numeric display mode to UNIT nn+1 digits ; setting for showing all or limited set of prefixes: PFX.All (display using prefix also available from HOME for numeric entry: p - n - μ - m - k - M - G - T for pico, nano, micro, milli, kilo, Giga, Tera) (UNIT __ TamNonReg menu)
Unit in the last place	Command	ULP?	ULP?	Minimum difference to next or previous machine representable real, as power of ten
Unit vector	Command	UNITV	UNITV	Unit vector for complex number or matrix
Unsigned	Setting (pgm)	UNSIGN	UNSIGN ()	Set unsigned mode for shortint (SBI depends on SBint)
Unsigned 16 bits	Setting (pgm)	U16		Shortcut to set word size to 16 bits unsigned
Unsigned 32 bits	Setting (pgm)	U32		Shortcut to set word size to 32 bits unsigned
Unsigned 6 bits	Setting (pgm)	U06		Shortcut to set word size to 6 bits unsigned
Unsigned 64 bits	Setting (pgm)	U64		Shortcut to set word size to 64 bits unsigned
Unsigned 8 bits	Setting (pgm)	U08		Shortcut to set word size to 8 bits unsigned
Up	Arrow	↑		Move up (navigation) or arrow character (alpha selection menus)
Up	Symbol	↑		Move up (navigation) or arrow character (alpha selection menus) (8593)
Upper limit of integration	Variable (real)	↑Lim	↑Lim	Upper limit of integration (reserved real variable) (↑Lim :)
Upper quantile	Command	×Q3	×Q3	Upper quantile for both X and Y (Q _{3 x} ; Q _{3 y} = (2 stack levels))
Upsilon	Character	Υ		Greek character Υ (933)
Upsilon dialytika	Character	ÿ		Greek character ÿ (939)
upsilon dialytika lowercase	Character	ÿ		Greek character ÿ (971)
upsilon dialytika tonos lowercase	Character	ÿ		Greek character ÿ (944)
upsilon lowercase	Character	υ		Greek character υ (965)
upsilon tonos lowercase	Character	ύ		Greek character ύ (973)

FullName	Type	Label	Catalog	Description
USA formatting	Setting (pgm)	USA	SETUSA	Set to USA regional formats (date, time, calendar, number formatting) (First Gregorian day set: 9/14/1752)
USB Power	Flag (system)	USB		Calculator is connected to USB power
User mode	Character	U		Character U (9260)
USER mode	Setting	USER		Set USER mode
USER mode	Flag	USER		Set for USER mode
V	Character	V		Character V (86)
v lowercase	Character	v		Character v (118)
V47 keyboard layout	Layout (SIM)	V47		V47: Exp Vintage 2 shifts TopR -+x/ L ; V = vintage
Vacuum electric permittivity	Constant (#60)	ϵ_0		epermt.vac $\epsilon_0 = +8.8541878128 \times 10^{-12}$ (As/Vm)
Vacuum magnetic permeability	Constant (#64)	μ_0		mpermb.vac $\mu_0 = +1.25663706212 \times 10^{-6}$ (Vs/Am)
VAR	MENU (TAM ; ASM)	VAR	VAR	Presented in TAM menus for commands accessing variables (CAT.VARS.+ menu ; Type characters 1-2 to search)
Variable menu	MENU (item ; PEM)	VarMNU	VARMNU	Create variable menu (VARMNU __ TamLbl(Alpha) menu)
Variable Menu Displayed	Flag (system)	VMDISP		Variable menu is displayed
Variables	MENU	VARS	VARS	Auto-generated catalog of variables
Vector angle	Function (dyadic)	V_{\angle}	V_{\angle}	Angle between two vectors (2D or 3D)
Version	Command (nonpgm)	VERS?	VERS?	Show firmware version (Firmware version)
View	Command	VIEW	VIEW	View register or variable (VIEW __ Tam menu)
Volume	Setting (pgm)	VOL	VOL ₁₁	Set audio volume (0-11) (VOL _ TamNonReg menu)
Volume conversion	MENU	Volume:	Volume:	Convert between units of volume
Volume down	Setting	VOL↓	VOL↓ ₁₁	Volume down (0-11)
Volume of ideal gas	Constant (#53)	V_m		volume.gas $V_m = +2,241396954501413773501110288675056 \times 10^{-2}$ (m ³ /mol)
Volume up	Setting	VOL↑	VOL↑ ₁₁	Volume up (0-11)
Volume?	Command	VOL?	VOL?	Audio volume (0-11)
Von Klitzing constant	Constant (#40)	R_K		c.klitzing $R_K = +2,581280745930450666004551670608744 \times 10^4$ (Ω)
W	Character	W		Character W (87)
W circumflex	Character	Ŵ		International character Ŵ (372)
w circumflex lowercase	Character	ŵ		International character ŵ (373)
w lowercase	Character	w		Character w (119)
W to hp _E	Function (linked ; monadic)	W→hp _E		Convert Watt to electrical horsepower (hp _E :)
W to hp _M	Function (linked ; monadic)	W→hp _M		Convert Watt to metric horsepower (hp _M :)
W to hp _{UK}	Function (linked ; monadic)	W→hp _{UK}		Convert Watt to UK horsepower (hp _{UK} :)
W _m	Function (tbd)	W _m	W _m	Lambert's W function (negative branch ; m = minus)
W _p	Function (tbd)	W _p	W _p	Lambert's W function (principal branch)
Weekday	Function (monadic)	WDAY	WDAY	Show weekday for date (Weekday)
Weibull (inverse)	Function (tbd)	Weibl ⁻¹	Weibl ⁻¹	Weibull probability inverse function
Weibull cdf (lower)	Function (tbd)	Weibl _∞	Weibl _∞	Weibull cumulative distribution (lower tail)
Weibull cdf (upper)	Function (tbd)	Weibl _∞	Weibl _∞	Weibull cumulative distribution (upper tail)
Weibull distribution	MENU	Weibl:	Weibl:	Weibull probability distribution (RegI = k = shape ; RegJ = λ = scale (lifetime))
Weibull pdf	Function (tbd)	Weibl _p	Weibl _p	Weibull probability density function
Weighted mean	Command	\bar{x}_w	\bar{x}_w	Weighted means of x with weight y ($\bar{x}_w =$)
Weighted population standard deviation	Command	σ_w	σ_w	Weighted population standard deviation ($\sigma_w =$)
Weighted population standard deviation	Command	s_w	s_w	Weighted population standard deviation ($s_w =$)
Wh to J	Function (linked ; monadic)	Wh→J		Convert Watt-hour to Joule (J:)
Who	Command (nonpgm)	WHO?	WHO?	Show calculator development team names (Team names)
Word size	Setting (pgm)	WSIZE	WSIZE ₆₄	Set word size for shortint (WSIZE __ TamNonReg menu ; Info : SBI depends on SBit)
Word size 16 bits	Setting (pgm)	16-BIT	16-BIT ()	Set word size to 16 bits for shortint
Word size 32 bits	Setting (pgm)	32-BIT	32-BIT ()	Set word size to 32 bits for shortint
Word size 64 bits	Setting (pgm)	64-BIT	64-BIT (•)	Set word size to 64 bits for shortint

FullName	Type	Label	Catalog	Description
Word size 8 bits	Setting (pgm)	8-BIT	8-BIT ()	Set word size to 8 bits for shortint
Word size?	Command	WSIZE?	WSIZE?	Word size for short integers ; set by WSIZE
Word swap	Function (monadic)	W.SWP	W.SWP	Swap words
WP43 keyboard layout	Layout	WP43		WP43 Pilot: Final compatibility layout
Wrap (matrix edit)	Setting	WRAP	M.WRAP (•)	Matrix edit in wrapping mode (SBI depends on SBmx)
Write program	Command	WRITEP	WRITEP	Write program to file <program.p47> in FAT (WRITEP _ TamLabel ; DMCP : File save dialog (PROGRAMS) ; Inverse : READP)
W ⁻¹	Function (tbd)	W ⁻¹	W ⁻¹	Inverse of W _p (≥ -1)
X	Character	X		Character X (88)
X approximates?	Function (monadic)	x≈ ?	x≈ ?	X approximates? (rounded values are equal) (x≈ ? __ TamCmp menu)
X Balanced	Function (monadic)	X → BAL	X → BAL	Create balanced 3 phase quantities by pushing onto stack X * a, and then X * a * a
x bar	Character	x̄		Character x̄ (888)
X equals -0?	Function (monadic)	x=-0 ?	x=-0 ?	X equals -0? (shortint 1COMPL or SIGNMT) (x=-0 ? __ TamCmp menu)
X equals +0?	Function (monadic)	x=+0 ?	x=+0 ?	X equals +0? (shortint 1COMPL or SIGNMT) (x=+0 ? __ TamCmp menu)
X equals?	Function (monadic)	x= ?	x= ?	X equals? (x= ? __ TamCmp menu)
X greater or equal?	Function (monadic)	x≥ ?	x≥ ?	X greater or equal? (x≥ ? __ TamCmp menu)
X greater?	Function (monadic)	x> ?	x> ?	X greater? (x> ? __ TamCmp menu)
x hat	Character	x̂		Character x̂ (889)
x hat	Command	x̂	x̂	Estimation of x
X less or equal?	Function (monadic)	x≤ ?	x≤ ?	X less or equal? (x≤ ? __ TamCmp menu)
X less?	Function (monadic)	x< ?	x< ?	X less? (x< ? __ TamCmp menu)
x lowercase	Character	x		Character x (120)
X not equal?	Function (monadic)	x≠ ?	x≠ ?	X not equal? (x≠ ? __ TamCmp menu)
X to alpha	Function (monadic)	x→α	x→α	Convert character code in X to alpha character (code value pushed to Y)
X to date	Function (monadic)	x→DATE	x→DATE	Convert date input number YYYY-MM-DD or DD.MM.YYYY or MM/DD/YYYY to date according to DISP or CLK format settings (Weekday)
x under root	Character	x̄		Character x̄ (895)
X-register	Command (TAM)	X		Presented in TAM menus for commands accessing stack level X
XEQ XEQM01	Command	XEQM01	XEQM01	Execute XEQM01 (HELP!!)
XEQ XEQM02	Command	XEQM02	XEQM02	Execute XEQM02 (BATPLT)
XEQ XEQM03	Command	XEQM03	XEQM03	Execute XEQM03 (MP2203)
XEQ XEQM04	Command	XEQM04	XEQM04	Execute XEQM04 (MP2281)
XEQ XEQM05	Command	XEQM05	XEQM05	Execute XEQM05 (MP3217)
XEQ XEQM06	Command	XEQM06	XEQM06	Execute XEQM06 (CUBES)
XEQ XEQM07	Command	XEQM07	XEQM07	Execute XEQM07 (GUDERM)
XEQ XEQM08	Command	XEQM08	XEQM08	Execute XEQM08 (PYTHAG)
XEQ XEQM09	Command	XEQM09	XEQM09	Execute XEQM09 (PLTPRIM)
XEQ XEQM10	Command	XEQM10	XEQM10	Execute XEQM10 (58TESTS)
XEQ XEQM11	Command	XEQM11	XEQM11	Execute XEQM11 (SINC_PI)
XEQ XEQM12	Command	XEQM12	XEQM12	Execute XEQM12 (BINET)
XEQ XEQM13	Command	XEQM13	XEQM13	Execute XEQM13 (TRAPZ)
XEQ XEQM14	Command	XEQM14	XEQM14	Execute XEQM14 (PLTFOR)
XEQ XEQM15	Command	XEQM15	XEQM15	Execute XEQM15 (X15)
XEQ XEQM16	Command	XEQM16	XEQM16	Execute XEQM16 (BINETF)
XEQ XEQM17	Command	XEQM17	XEQM17	Execute XEQM17 (RANDOM)
XEQ XEQM18	Command	XEQM18	XEQM18	Execute XEQM18 (TEST)
XEQM	MENU	XEQM	XEQM	Menu of predefined XEQC-functions
Xi	Character	Ξ		Greek character Ξ (926)
xi lowercase	Character	ξ		Greek character ξ (958)
XOR	Function (dyadic)	XOR	XOR	Logical exclusive OR

FullName	Type	Label	Catalog	Description
x_{SUM}	Command	x_{SUM}	x_{SUM}	Return Σx and Σy in X and Y respectively (Σ_x ; Σ_y = (2 stack levels))
xth root	Character	$\sqrt[x]{y}$		Character $\sqrt[x]{y}$ (8732)
xth root	Function (dyadic)	$\sqrt[x]{y}$	$\sqrt[x]{y}$	Xth root of Y
XXEQ	MENU	XXEQ	XXEQ	XXEQ menu (Longpress [XEQ])
Y	Character	Y		Character Y (89)
Y acute	Character	Ÿ		International character Ÿ (221)
y acute lowercase	Character	ÿ		International character ÿ (253)
y bar	Character	ȳ		Character ȳ (563)
Y circumflex	Character	ÿ		International character ÿ (374)
Y diaeresis	Character	ÿ		International character ÿ (376)
y diaeresis lowercase	Character	ÿ		International character ÿ (255)
y hat	Command	ŷ	ŷ	Estimation of y
y hat (circumflex lowercase)	Character	ŷ		Character ŷ (375)
y lowercase	Character	y		Character y (121)
y to the power x	Function (dyadic)	y^x	y^x	Raise value in the Y-register to the power in the X-register
y under root	Character	$\sqrt[y]{x}$		Character $\sqrt[y]{x}$ (562)
Y-register	Command (TAM)	Y		Presented in TAM menus for commands accessing stack level Y
$Y_v(x)$	Command	$Y_v(x)$	$Y_v(x)$	Bessel function of the 2nd kind and order y
yd. to m	Function (linked ; monadic)	yd.→m		Convert yard to meter (m:)
Year	Command	YEAR	YEAR	Year (of date)
Year month day	Setting (pgm)	YMD	YMD (*)	Date display mode YYYY-MM-DD (YYYY-MM-DD)
Year month day	Flag (system)	YMD		Date display mode YYYY-MM-DD (YYYY-MM-DD)
year to s	Function (linked ; monadic)	year→s		Convert year to second (s:)
Yen	Character	¥		Character ¥ (165)
yīn to m	Function (linked ; monadic)	yīn→m		Convert yīn to meter (m:)
Z	Character	Z		Character Z (90)
Z acute	Character	Ẑ		International character Ẑ (377)
z acute lowercase	Character	ẑ		International character ẑ (378)
Z caron	Character	Ž		International character Ž (381)
z caron lowercase	Character	ž		International character ž (382)
Z dot	Character	Ẓ		International character Ẓ (379)
z dot lowercase	Character	ẓ		International character ẓ (380)
z lowercase	Character	z		Character z (122)
Z-register	Command (TAM)	Z		Presented in TAM menus for commands accessing stack level Z
Z^Y modulo X	Function (triadic)	^MOD	^MOD	Z^Y modulo X
Zeta	Character	Ζ		Greek character Ζ (918)
zeta lowercase	Character	ζ		Greek character ζ (950)
zhàng to m	Function (linked ; monadic)	zhàng→m		Convert zhàng to meter (m:)
Zoom (plot)	Command	ZOOM		Zoom (plot)
Zoom x-axis	Command	ZOOMx ₀		Cycle through a preset list of 4 zoom factors on the x-axis
Zoom y-axis	Command	ZOOMy ₀		Cycle through a preset list of 4 zoom factors on the y-axis
ZxY modulo X	Function (triadic)	*MOD	*MOD	Z x Y, modulo X
Γ function	Function (monadic)	Γ(x)	Γ(x)	Gamma function
Γ _{xy}	Function (dyadic)	Γ _{xy}	Γ _{xy}	Upper incomplete Gamma function
γ _{xy}	Function (dyadic)	γ _{xy}	γ _{xy}	Lower incomplete Gamma function
ζ(x)	Function (monadic)	ζ(x)	ζ(x)	Riemann's Zeta for real arguments
Σ+	Setting	Σ+		Assign Σ+ to Σ+ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as Σ+ (normal))

FullName	Type	Label	Catalog	Description
Σ CC	Setting	Σ CC		Assign CC to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [CC] (in reverse))
Σ DRG	Setting	Σ DRG		Assign DRG to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [DRG] (in reverse))
Σ g	Setting	Σ g		Assign [G] to $\Sigma+$ key for NORMAL as well as USER mode ; disables longpress on $\Sigma+$ and the use of this key as A in ASM 2-character search ; shift keys are not assignable to any key in USER mode (Shown in KEYMAP as [G] (in reverse))
Σ HOME	Setting	Σ HOME		Assign HOME to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [HOME] (in reverse))
Σ MyM	Setting	Σ MyM		Assign MyMenu to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [MyM] (in reverse))
Σ PRGM	Setting	Σ PRGM		Assign PRGM to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [PRGM] (in reverse))
Σ SNAP	Setting	Σ SNAP		Assign SNAP to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [SNAP] (in reverse))
Σ USER	Setting	Σ USER		Assign USER to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [USER] (in reverse))
$\Sigma\alpha$	Setting	$\Sigma\alpha$		Assign α to $\Sigma+$ key for NORMAL mode ONLY & deactivate USER mode (Shown in KEYMAP as [α] (in reverse))
χ^2 distribution	MENU	χ^2 :	χ^2 :	χ^2 distribution (RegI = ν = degrees of freedom)
$\chi^2_p(x)$	Function (tbd)	$\chi^2_p(x)$	$\chi^2_p(x)$	χ^2 probability density function
$\chi^2_{\Delta}(x)$	Function (tbd)	$\chi^2_{\Delta}(x)$	$\chi^2_{\Delta}(x)$	χ^2 cumulative distribution (lower tail)
$\chi^2_{\Delta}(x)$	Function (tbd)	$\chi^2_{\Delta}(x)$	$\chi^2_{\Delta}(x)$	χ^2 cumulative distribution (upper tail)