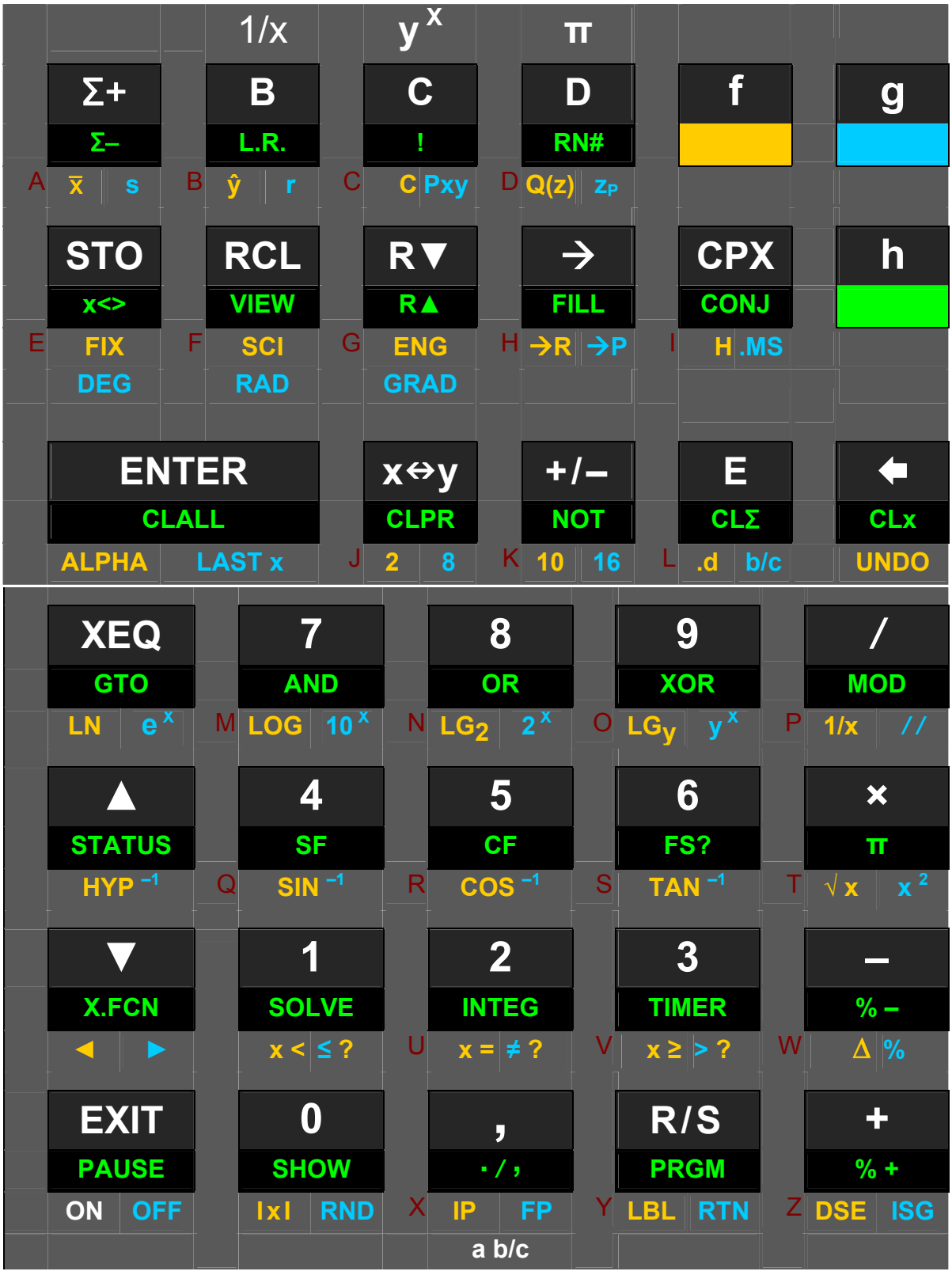
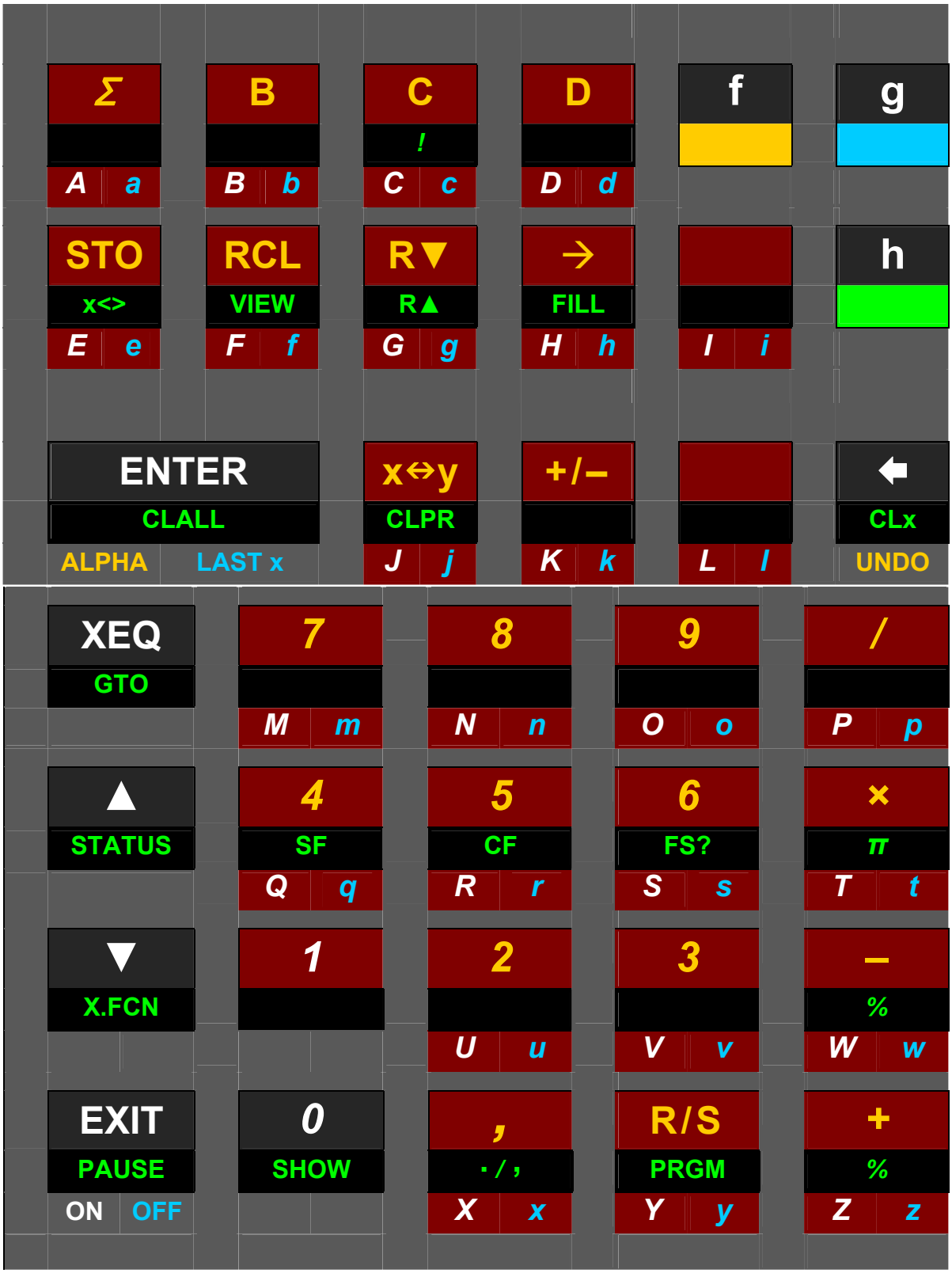


Keyboard layout:

- CPX may be combined with +, -, x, /, ±, x^2, √x, 1/x, //, Γ, IxI, RND, as well as e.g. (HYP) SIN, COS, TAN, LN, LG_y and their inverses. See the index for more.
- Modes are H.MS, 2, 8, 10, 16, .d, b/c, FIX, SCI, ENG, DEG, RAD, GRAD.
- → is combined with H, H.MS, DEG, RAD, GRAD, 2, 8, 10, 16 for conversions.
- The keys B, C, and D immediately call the respective user programs if existent.

Active operations in hexadecimal mode. ➡ is for addressing only (see below). The primary functions of the top left 6 keys are numeric input, so their default primary functions are accessed using f-shift. – In the other integer modes, the active keyboard will look alike, but the primary functions of the top left 6 keys will stay as they are in DECM.



Active operations in alpha mode. ➡ is for addressing only (see below). The primary function of most keys will be inserting a letter. The basic arithmetic keys, Σ, π, %, and "!" will do so, too. To reach the default primary functions, f-shift will be necessary wherever a letter stands next to a key.

ADDRESSING REGISTERS

1	User input	x= ? or any of the other comparisons			RCL, STO, VIEW, x≷, DSE, ISG, DSZ, ISZ, SF, CF, FS?, SB, CB, BS?, FIX, SCI, ENG			
	Display	OP _ (e.g. x > _) Alpha mode is set.			OP _ (e.g. RCL _)			
2	User input [1]	0 or 1	Y, Z, T, or L	ENTER↑	ENTER↑ [2]	*Number of register or flag or bit or decimals* [3]	→	
	Display	**OP 0** e.g. x ≤ 0	**OP *x*** e.g. x ≥ y	**OP r _**	**OP s _** Alpha mode is set.	**OP *nn*** e.g. SF 15	**OP → _**	
						(indirect addressing)		
3	User input			*Register no.* [3]	X, Y, Z, T, or L	ENTER↑	*Register number* [3]	
	Display	Compare ***x*** with the number in register **23**.		**OP r *nn*** e.g. x ≠ r23	**OP s *x*** e.g. STO sZ	**OP →s _** Alpha mode is set.	**OP → *nn*** e.g. SCI →03	
4	User input				Store ***x*** on stack level **z**.	X, Y, Z, T, or L	Choose scientific number display with the number of decimals specified in register **03**.	
	Display				Show the content of the register where **LASTx** is pointing to.	**OP →s *x*** e.g. VIEW →sL		

[1] For **RCL** and **STO**, an arithmetic operator (+, –, ×, / , ^) may precede step 2.

[2] For **RCL**, **STO**, **VIEW**, and **x<>** only.

[3] Register numbers may be 0 0 … 9 9, flag numbers as well, bit numbers 0 1 … 6 4., number of decimals 0 … 9 . For any of the lowest 10 registers, flags, or bits, you may enter e.g. 5 ENTER↑ instead of 0 5 .

ADDRESSING LABELS

<table>
<tr><td>1</td><td>User input</td><td colspan="3">GTO or XEQ , LBL, SOLVE, INTEG [4]</td></tr>
<tr><td></td><td>Display</td><td colspan="3">OP "_ (e.g. GTO _)
Alpha mode is set</td></tr>
<tr><td>2</td><td>User input</td><td>Label
+ ENTER↑ [5]</td><td>ENTER↑ →</td><td></td></tr>
<tr><td></td><td>Display</td><td>OP "name"
e.g. SLV"STF"</td><td>OP → _</td><td></td></tr>
<tr><td></td><td></td><td>Solve the function STF
(with STF keyed in).</td><td colspan="2">(indirect addressing)</td></tr>
<tr><td>3</td><td>User input</td><td colspan="2">ENTER↑</td><td>Register number</td></tr>
<tr><td></td><td>Display</td><td colspan="2">OP →s _
Alpha mode is set.</td><td>OP → nn
e.g. XEQ →44</td></tr>
<tr><td>4</td><td>User input</td><td colspan="2">X , Y , Z , T , or L</td><td>Execute the routine which's label is in register 44.</td></tr>
<tr><td></td><td>Display</td><td>Integrate the function which's label is on stack level y.</td><td>OP →s x
e.g. INT →sY</td><td></td></tr>
</table>

[4] **SOLVE** and **INTEG** will be displayed as SLV and INT, respectively. No indirect addressing with **LBL**.

[5] A label may consist of up to 3 alphanumeric characters. ENTER↑ is only needed if less than 3 characters are entered.

INDICATORS

There are a number of indicators signaling the mode the calculator is running in. The defaults DECM and DEG are not indicated.

Indicator	a	2	d	h	8	PRG
Set by operation	AON ENTER X.FCN	BINM	IDECM	HEXM	OCTM	PRGON
Cleared by operation	AOFF ENTER	DECM FRACM HEXM IDECM OCTM	BINM DECM FRACM HEXM OCTM	BINM DECM FRACM IDECM OCTM	BINM DECM FRACM HEXM IDECM	PRGOFF

Indicator	(DEG)	GRAD	HMS	RAD	/c
Set by operation	DEG	GRAD	HMSM >HMS TIMER	RAD ACOSH ASINH ATANH	FRACM, 2nd [,] in input (\HMS)
Cleared by operation	GRAD RAD ACOSH ASINH ATANH	DEG RAD ACOSH ASINH ATANH >HR	DECM >HR	DEG GRAD >HR	BINM DECM IDECM HEXM OCTM

Within integer modes, the annunciator “c” shows a carry bit set, and the word size and complement setting are shown in STATUS, as well as the status of the lowest flags.

Outside integer modes, an active timer application is indicated by T, the different date modes by D.MY or M.DY (default Y.MD is not indicated).

INDEX OF OPERATIONS

This lists all functions available on the 34S with the necessary keystrokes. Functions accessible via X.FCN will show up with their name unless specified differently explicitly. Generally, the operations will work like on the HP-42S, special bit and integer functions like on the HP-16C. Functions available here for the first time on an RPN calculator are highlighted. If no parameters are specified though required, they will be taken from the stack. Modes are abbreviated by their indicators. There, a backslash stands for "all but", so e.g. ABS works in all modes but alpha.

Name	Keys to press	Works in modes	Remarks
A … F	A etc.	h	The top left 6 keys will work for numeric input in hexadecimal mode by default.
ABS	f |x|	\a	CPX f |x| calculates $\sqrt{x^2+y^2}$.
ACOS	g COS^{-1}	DECM, H.MS	
ACOSH	g HYP^{-1} COS	DECM	
ALL	h X.FCN ...	DECM	
AND	h AND	2, 8, d, h	
AOFF	f ALPHA ENTER↑	a	Toggles alpha mode. *) Please see the table for register addressing above for details.
AON	f ALPHA ENTER↑	\a *)	
ASIN	g SIN^{-1}	DECM, H.MS	
ASINH	g HYP^{-1} SIN	DECM	
ASR	h X.FCN ...	2, 8, d, h	Shifts ***x*** right 1 bit, like in HP-16C.
ASRN	h X.FCN **ASRn**	2, 8, d, h	Shifts ***y*** right |***x***| bits. The operation will work like ***x*** consecutive ASRs.
ATAN	g TAN^{-1}	DECM, H.MS	
ATANH	g HYP^{-1} TAN	DECM	
A0	h L.R.	DECM	Calculates the parameters of the fit curve (through the data points accumulated) according to the model selected. In the linear model, A0 is the intercept and A1 the slope of the regression line.
A1	h L.R. x≷y		

Name	Keys to press	Works in modes	Remarks
BASE+	[+]	2, 8, d, h	
BASE–	[−]		
BASE×	[×]		
BASE/	[/]		
BASE+/–	[+/−]		
BC?	[h] [X.FCN] ...	2, 8, d, h	
BETA	[h] [X.FCN] ...	DECM	Calculates Euler's Beta function.
BINM	[f] [2]	\a	Sets binary mode.
BST	[▲]	PRG	Go 1 step back in program memory.
		\PRG	Go 1 step back without executing this step.
BS?	[h] [X.FCN] ...	2, 8, d, h	Works like **B?** in HP-16C.
CB	[h] [X.FCN] ...	2, 8, d, h	
CBC?	[h] [X.FCN] ...	2, 8, d, h	Carry bit clear?
CBS?	[h] [X.FCN] ...	2, 8, d, h	Carry bit set?
CCB	[h] [X.FCN] ...	2, 8, d, h	Clear carry bit.
CEIL	[h] [X.FCN] ...	DECM	Computes largest integer ≤ ***x***.
CF	[h] [CF] arg	All	See opportunities for "arg" in the table above.
CHS	[+/−]	\a	[CPX] [+/−] changes signs of ***x*** and ***y***.
CLALL	[h] [CLALL]	All	Global clear after confirmation.
CLPR	[h] [CLPR]	PRG	Clears current program after confirmation.
CLRG	[h] [X.FCN] ...	All	Clears all general purpose registers.
CLSTK	[0] [h] [FILL]	All	
CLX	[h] [CLx]	All	[CPX] [h] [CLx] clears **x** and **y**.
CLΣ	[h] [CLΣ]	DECM	
COMB	[f] [Cx,y]	DECM	

Name	Keys to press	Works in modes	Remarks
COMPLEX	CPX ...	DECM	Indicates complex operations, acting on ***x*** and ***y***, where **x** contains the real part and **y** the imaginary of the complex number. This key may be combined with any function which's name is printed in *italics* in this table.
CONJ	h CONJ	DECM	Changes the sign of ***y***.
CORR	g r	DECM	
COS	f COS	DECM, H.MS	
COSH	f HYP COS	DECM	
DATE	h X.FCN ...	DECM	Adds a number of days in ***x*** on a date in ***y*** and displays the resulting date including the respective weekday (Sunday = 7). This function works like in HP-12C.
DDAYS	h X.FCN ...	DECM	Calculates the number of days between 2 dates ***x*** and ***y***. Function works like in 12C.
DECM	f .d	\a	H.MS data in **x** will be converted to decimal.
DEG	g DEG	DECM	
DENMAX	h X.FCN */c* n	DECM	Sets the maximum denominator for fractions.
DSE	f DSE *reg*	PRG	See opportunities for "reg" in the table above.
DSZ	h X.FCN ...	PRG	
D.MY	h X.FCN ...	DECM	Sets the format for date calculations.
D→R	h X.FCN ...	DECM	Assumes **x** containing radians and converts them to degrees. Mode is kept constant.
EEX	E	DECM	
EMGAM	h X.FCN ...	DECM	Displays the Euler-Mascheroni constant.
ENG	f ENG *arg*	DECM	See opportunities for "arg" in the table above.
ENTER↑	ENTER↑	All	
EXIT	EXIT	All	Exits X.FCN and any other menus or functions with pending input.
EXPF	h X.FCN ...	DECM	Selects the exponential curve fit model.
E↑X	g e^x	DECM	

Name	Keys to press	Works in modes	Remarks
FB	h X.FCN ...	2, 8, d, h	Flip bit (i.e. invert the bit specified).
FCB	h X.FCN ...	2, 8, d, h	Flip carry bit.
FCSTX	h X.FCN ...	DECM	FCSTX (*FCSTY*) predicts x (*y*) for a given y (*x*) according to the curve fit model chosen. See A0 and A1 for more.
FCSTY	f ŷ		
FC?	h X.FCN ...	All	
FF	h X.FCN ...	All	Flip flag (i.e. invert the flag specified).
FILL	h FILL	All	Copies ***x*** in **y**, **z**, and **t**.
FIX	f FIX *arg*	DECM	See opportunities for “arg” in the table above.
FLOAT	f .d	\a	Works like DECM.
FLOOR	h X.FCN ...	DECM	Computes the smallest integer ≤ ***x***.
FP	g FP	DECM	
FRACM	g b/c	DECM	Sets fraction mode like in HP-32SII.
FS?	h FS? *arg*	All	Like **F?** in HP-16C. See opportunities for “arg” in the table above.
GAMMA	h X.FCN ...	DECM	
GRAD	g GRAD	DECM	
GSB	h XEQ *label*	PRG	Works like XEQ.
GTO	h GTO *label*	PRG \PRG	Like in HP32S
	h GTO . *label*	\PRG	
	h GTO . .		
HEXM	g 16	\a	Sets hexadecimal mode.
HMSM	g H.MS	DECM	Sets H.MS mode.
HMS+	+	H.MS	
HMS–	–	H.MS	
IDECM	f 10	\a	Sets integer decimal mode.

Name	Keys to press	Works in modes	Remarks
INTEG	h INTEG *label*	DECM	Parameters will be transferred like in 15C.
IP	f IP	DECM	
ISG	g ISG *reg*	PRG	See opportunities for “reg” in the table above.
ISZ	h X.FCN ...	PRG	
LASTX	g LASTx	All	CPX g LASTx recalls **x** and **y**.
LBL	f LBL *label*	PRG	
LINF	h X.FCN ...	DECM	Selects the linear curve fit model.
LJ	h X.FCN ...	2, 8, d, h	
LN	f LN	DECM	
LNBETA	h X.FCN ...	DECM	Calculates the logarithm of BETA.
LNGAMMA	h X.FCN ...	DECM	Calculates the logarithm of GAMMA.
LOG	f LOG	DECM	
LOGF	h X.FCN ...	DECM	Selects the logarithmic curve fit model.
LOGY	f LGy	DECM	Calculates the logarithm for base ***y***.
LOG2	f LG2	DECM	Calculates the logarithm for base 2.
MASKL	h X.FCN ...	2, 8, d, h	
MASKR	h X.FCN ...	2, 8, d, h	
MAX	h X.FCN ...	\a	Takes the maximum of ***x*** and ***y***.
MEAN	f $\bar{x}$	DECM	
MIN	h X.FCN ...	\a	Takes the minimum of ***x*** and ***y***.
MIRROR	h X.FCN ...	2, 8, d, h	Reflects bit patterns (e.g. 000101 → 101000)
MOD	h MOD	\a	
M.DY	h X.FCN ...	DECM	Sets the format for date calculations.

Name	Keys to press	Works in modes	Remarks
N	h X.FCN ...	DECM	Recalls the number of data points accumulated. Necessary for basic statistics.
NAND	h X.FCN ...	2, 8, d, h	
NBITS	h X.FCN **#B**	2, 8, d, h	Like **#B** on HP-16C.
NOP	h X.FCN ...	PRG	
NOR	h X.FCN ...	2, 8, d, h	
NOT	h NOT	2, 8, d, h	
OCTM	g 8	\a	Sets octal mode.
OFF	g OFF	\PRG	
	h X.FCN ...	PRG	
ON	ON	Calc. off	
	h X.FCN ...	PRG	
OR	h OR	2, 8, d, h	
PAUSE	h PAUSE	PRG	
PERM	g Py,x	DECM	
PGOFF	h PRGM	PRG	Toggles programming mode.
PGON		\PRG	
PI	h π	DECM	
	D	DECM	As long as no reassignment took place.
PWRF	h X.FCN ...	DECM	Selects the power curve fit model.
QZ	f Qz	DECM	Like in HP-32E and HP-21S
RAD	g RAD	DECM	
RAND	h RN#	\a	

Name	Keys to press	Works in modes	Remarks
RCL	RCL *reg*	\h	See RCL+ for more details.
	f RCL *reg*	h	
RCLWS	h X.FCN ...	2, 8, d, h	Recalls the word size set.
RCL+	RCL + *reg*	\h (needs f in hex mode)	See opportunities for "reg" in the table above. CPX RCL recalls the register specified and the next adjacent register to **x** and **y**.
RCL–	RCL − *reg*		
RCL×	RCL × *reg*		
RCL/	RCL / *reg*		
RDN	R↓	All	
RDX,	h ./,	DECM	Toggles radix mark.
RDX.			
REAL	f .d	\a	Works like DECM.
RJ	h X.FCN ...	2, 8, d, h	Works in analogy to LJ.
RL	h X.FCN ...	2, 8, d, h	
RLC	h X.FCN ...	2, 8, d, h	
RLCN	h X.FCN **RLCn**	2, 8, d, h	
RLN	h X.FCN **RLn**	2, 8, d, h	
RND	g RND	DECM	
RR	h X.FCN ...	2, 8, d, h	
RRC	h X.FCN ...	2, 8, d, h	
RRCN	h X.FCN **RRCn**	2, 8, d, h	
RRN	h X.FCN **RRn**	2, 8, d, h	
RTN	g RTN	PRG	
RUP	h R↑	All	
R/S	R/S	\PRG, \T	
		T	Starts/stops incrementing the timer.

Name	Keys to press	Works in modes	Remarks
R→D	h X.FCN ...	DECM	Assumes **x** containing degrees and converts them to radians. Mode is kept constant.
SB	h X.FCN ...	2, 8, d, h	
SCB	h X.FCN ...	2, 8, d, h	Set carry bit.
SCI	f SCI *arg*	DECM	See opportunities for "arg" in the table above.
SDEV	g s	DECM	
SERR	h X.FCN ...	DECM	Calculates $\frac{SDEV}{\sqrt{N}}$.
SEED	STO h RN#	DECM	
SF	h SF *arg*	All	See opportunities for "arg" in the table above.
SHOW	h SHOW	All	
SIGMA	h X.FCN ...	DECM	Calculates $SDEV \cdot \sqrt{\frac{N}{N-1}}$.
SIN	f SIN	DECM, H.MS	
SINH	f HYP SIN	DECM	
SL	h X.FCN ...	2, 8, d, h	Shifts ***x*** left 1 bit, like in HP-16C.
SLN	h X.FCN **SLn**	2, 8, d, h	Shifts ***y*** left I***x***I bits. The operation will work like ***x*** consecutive SLs.
SOLVE	h SOLVE *label*	DECM	
SQRT	f √x	DECM	
SR	h X.FCN ...	2, 8, d, h	Shifts ***x*** right 1 bit, like in HP-16C.
SRN	h X.FCN **SRn**	2, 8, d, h	Shifts ***y*** right I***x***I bits. The operation will work like ***x*** consecutive SRs.
SST	▼	PRG	Go 1 step forward in program memory.
		\PRG	Go 1 step forward executing this step.
STATUS	h STATUS	All	

Name	Keys to press	Works in modes	Remarks
STO	STO *reg*	\h	See STO+ for more details.
	f STO *reg*	h	
STOP	R/S	PRG	
STO+	STO + *reg*	\h (needs f in hex mode)	See opportunities for "reg" in the table above. CPX STO stores ***x*** and ***y*** into the register specified and the next adjacent register.
STO–	STO − *reg*		
STO×	STO × *reg*		
STO/	STO / *reg*		
SUM	RCL Σ+	DECM	
TAN	f TAN	DECM, H.MS	
TANH	f HYP TAN	DECM	
TIMER	h TIMER *reg*	\T	Enters the timer application (sets indicator T, and HMS if not set yet). Clears the register specified.
TIMEX	h TIMER	T	Leaves the timer application (clears indicator T, but H.MS stays on).
UNDO	f UNDO	All	
UNSIGN	h X.FCN ...	All	
VIEW	h VIEW *reg*	All	See opportunities for "reg" in the table above.
W	h X.FCN ...	DECM	Calculates Lambert's W for a given x ≥ -1/e
WINV	h X.FCN **W** $^{-1}$	DECM	Inverts W, i.e. calculates x for a given W (≥ -1).
WMEAN	h X.FCN ...	DECM	Calculates the weighted mean.
WSIZE	h X.FCN ...	All	Sets the word size (up to 64 bits).
XEQ	XEQ *label*	PRG	Calls the respective subroutine.
		\PRG	Executes the respective program.
XNOR	h X.FCN ...	2, 8, d, h	
XOR	h XOR	2, 8, d, h	
X!	h !	DECM	

Name	Keys to press	Works in modes	Remarks
X.FCN	h X.FCN	DECM	Calls the catalog of extra real functions and sets alpha mode to allow for keying in names of operations. ▲ and ▼ browse the catalog. Choose the operation displayed by pressing ENTER↑ . – CPX h X.FCN calls the catalog of extra complex functions.
		2, 8, d, h	Calls the catalog of extra integer functions. See above for more.
X<>	h x≷ *reg*	All	See opportunities for "reg" in the table above. CPX h x≷ exchanges ***x*** and ***y*** with the register specified and the next adjacent register.
X<>Y	x≷y	All	Performs **Re <> Im** if a complex operation was executed before. CPX x≷y exchanges ***x*** and ***y*** with ***z*** and ***t***.
X< ?	f x < ? *arg*	\a	See opportunities for "arg" in the table above.
X≤ ?	g x ≤ ? *arg*		
X= ?	f x = ? *arg*		
X≠ ?	g x ≠ ? *arg*		
X≥ ?	f x ≥ ? *arg*		
X> ?	g x > ? *arg*		
X↑2	g x^2	\a	
Y.MD	h X.FCN ...	DECM	Sets the format for date calculations.
Y↑X	g y^x	DECM	
	C	DECM	As long as no reassignment took place.
ZETA	h X.FCN ...	DECM	Calculates Riemann's Zeta.
ZP	g zP	DECM	Like in $\mathbf{Q^{-1}}$ in HP-32E and $\mathbf{z_P}$ in HP-21S
0, 1	0 , 1	All	
2 … 7	2 …7	\2	
8, 9	8 , 9	\2, \8	

Name	Keys to press	Works in modes	Remarks
1/X	f 1/x	DECM	
	B	DECM	As long as no reassignment took place.
1CPL	h X.FCN ...	All	Like 1's complement in HP-16C.
2CPL	h X.FCN ...	All	Like 2's complement in HP-16C.
2↑X	g 2^x	DECM	
10↑X	g 10^x	DECM	
[.] or [,]	,	DECM	Inserts the radix mark as selected.
		a	Inserts a point (if RDX**.**) or comma (if RDX**,**).
	h ./,	a	Inserts a comma (if RDX**.**) or point (if RDX**,**).
[ ] or [/]	,	/c	
[°]	,	H.MS	
+	+	DECM	
–	–		
×	×		
/	/		
+/–	+/–		
/ /	g //	DECM	Calculates $\left(\frac{1}{x}+\frac{1}{y}\right)^{-1}$.
%	g %	DECM	
%CH	f Δ%	DECM	
%+	h %+	DECM	Adds a markup of ***x*** %.
%–	h %–	DECM	Subtracts a discount of ***x*** %.
Σ+	Σ+	DECM	
Σ–	h Σ–	DECM	

Name	Keys to press	Works in modes	Remarks
ΣLNX	h X.FCN ...	DECM	Recalls the respective sum. These sums are necessary for the other curve fitting models beyond pure linear. See below for more.
ΣLNXY	h X.FCN ...		
ΣLNX2	h X.FCN ...		
ΣLNY	h X.FCN ...		
ΣLNY2	h X.FCN ...		
ΣX	h X.FCN ...	DECM	Recalls the respective sum. These sums are necessary for basic statistics and linear curve fitting. Calling them by name greatly enhances readability of programs. These 11 statistical sums (in total) shall be stored in registers 88 through 99.
ΣXY	h X.FCN ...		
ΣX2	h X.FCN ...		
ΣY	h X.FCN ...		
ΣY2	h X.FCN ...		
→BIN	→ f 2	\a	Shows ***x*** in binary or decimal representation, respectively, until the next command is executed. Mode is kept constant.
→DEC	→ f 10		
→DEG	→ g DEG	DECM	Assumes **x** containing angles in current mode and converts them to degrees or gon, respectively. Mode is kept constant.
→GRAD	→ g GRAD		
→HEX	→ g 16	\a	Works like →BIN, but hexadecimal.
→HMS	→ g H.MS	DECM	Assumes **x** containing *decimal* hours or degrees and displays them in the format HHH.MMSS.
→HR	→ f H	H.MS	Takes the hours or degrees in **x** and displays them as decimal numbers.
→OCT	→ g 8	\a	Works like →BIN, but octal.
→POL	g ▶P	DECM	Assumes **x** and **y** containing the coordinates x and y and converts them to r and ϑ.
→RAD	→ g RAD	DECM, H.MS	Works like →DEG, but converts to radians.
→REC	f ▶R	DECM	Assumes **x** and **y** containing the coordinates r and ϑ and converts them to x and y.
	f ◀|	2	Shift the display window like in HP-16C.
	g |▶		

Functions on the waitlist (personal priorities given by W):

1. **Alpha** handling (partially included in the index already)
2. **SOLVE** (included in the index already but not implemented yet)
3. **TIMER** (accuracy down to 0.1s is sufficient, we can't press keys more precisely – included in the index already but not implemented yet)
4. **INTEG**RATE (included in the index already but not implemented yet)
5. **T**, **CHISQ**, and **F** distributions (in this order) and their inverses
6. **Julian Calendar** calculations
7. Explicit handling of **carry** bits (included in the index already but may be dropped if bytes are needed)
8. Euler's **BETA** function, Lambert's **W** and Riemann's **Zeta** (included in the index already but may be dropped if bytes are needed)
9. Explicit handling of **overflow** bits (doubt the use)
10. Stranger additional flag and bit handling like **FC?C**, **BS?S**, etc., also for carry and overflow (doubt the use – the user may program this if wanted)
11. **x^y** (the user may reach this by pressing x<>y and y^x)
12. **%T** (i.e. y ENTER x / 100 *)

Edition	Date	Remarks
1	9.12.08	Start
1.1	15.12.08	Added the table of indicators; added NAND, NOR, XNOR, RCLWS, STOWS, //, N, SERR, SIGMA, < and >; deleted HR, INPUT, 2 flag commands, and 2 conversions; extended explanations for addressing and COMPLEX & …; put XOR on the keyboard; corrected errors.
1.2	4.1.09	Added ASRN, CBC?, CBS?, CCB, SCB, FLOAT, MIRROR, SLN, SRN, >BIN, >DEC, >HEX, >OCT, BETA, D>R, DATE, DDAYS, D.MY, M.DY, Y.MD, CEIL, FLOOR, DSZ, ISZ, D>R, R>D, EMGAM, GSB, LNBETA, LNGAMMA, MAX, MIN, NOP, REAL, RJ, W and WINV, ZETA, %+ and %-; renamed the top left keys B, C, and D, and bottom left EXIT.