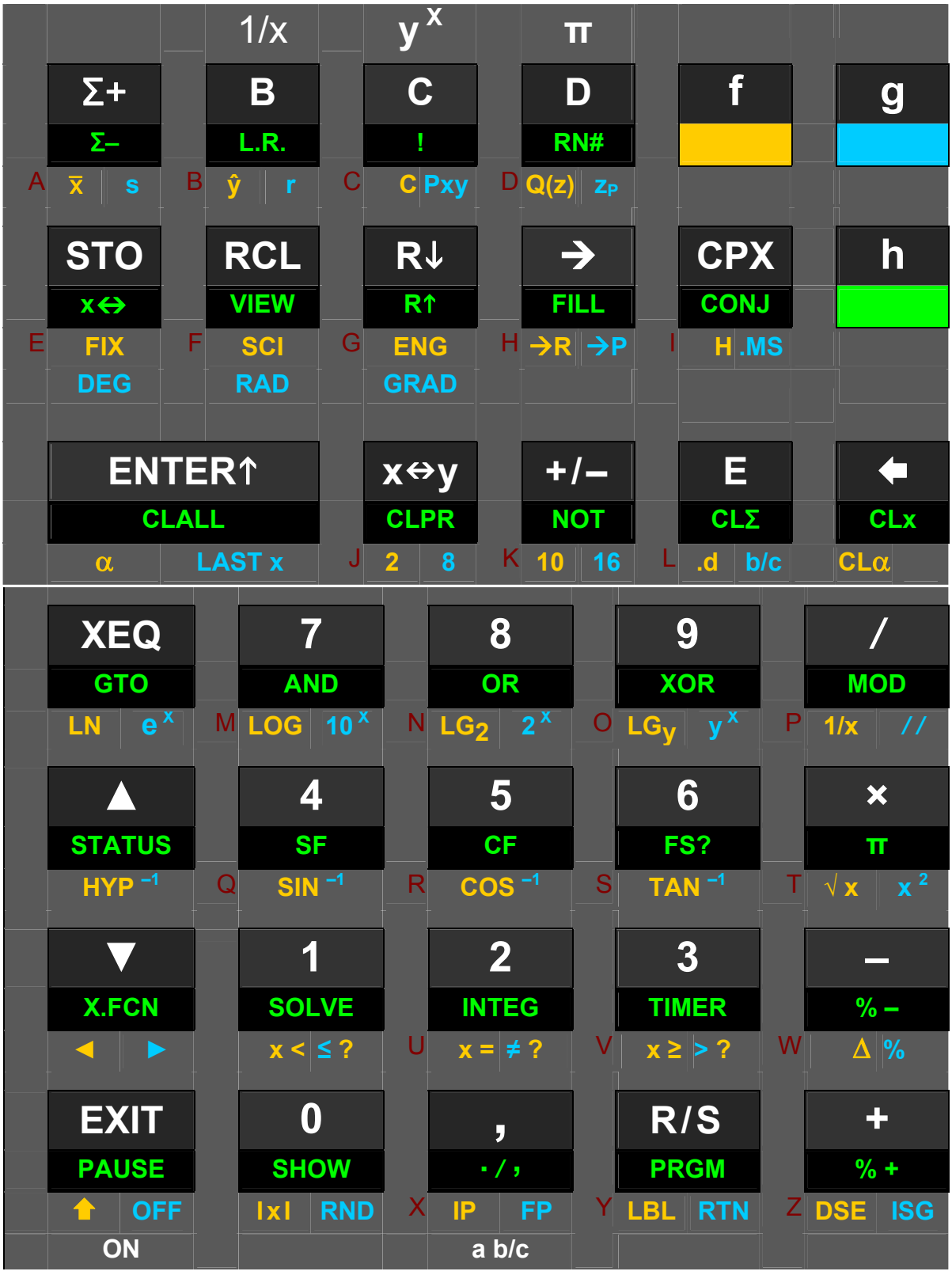
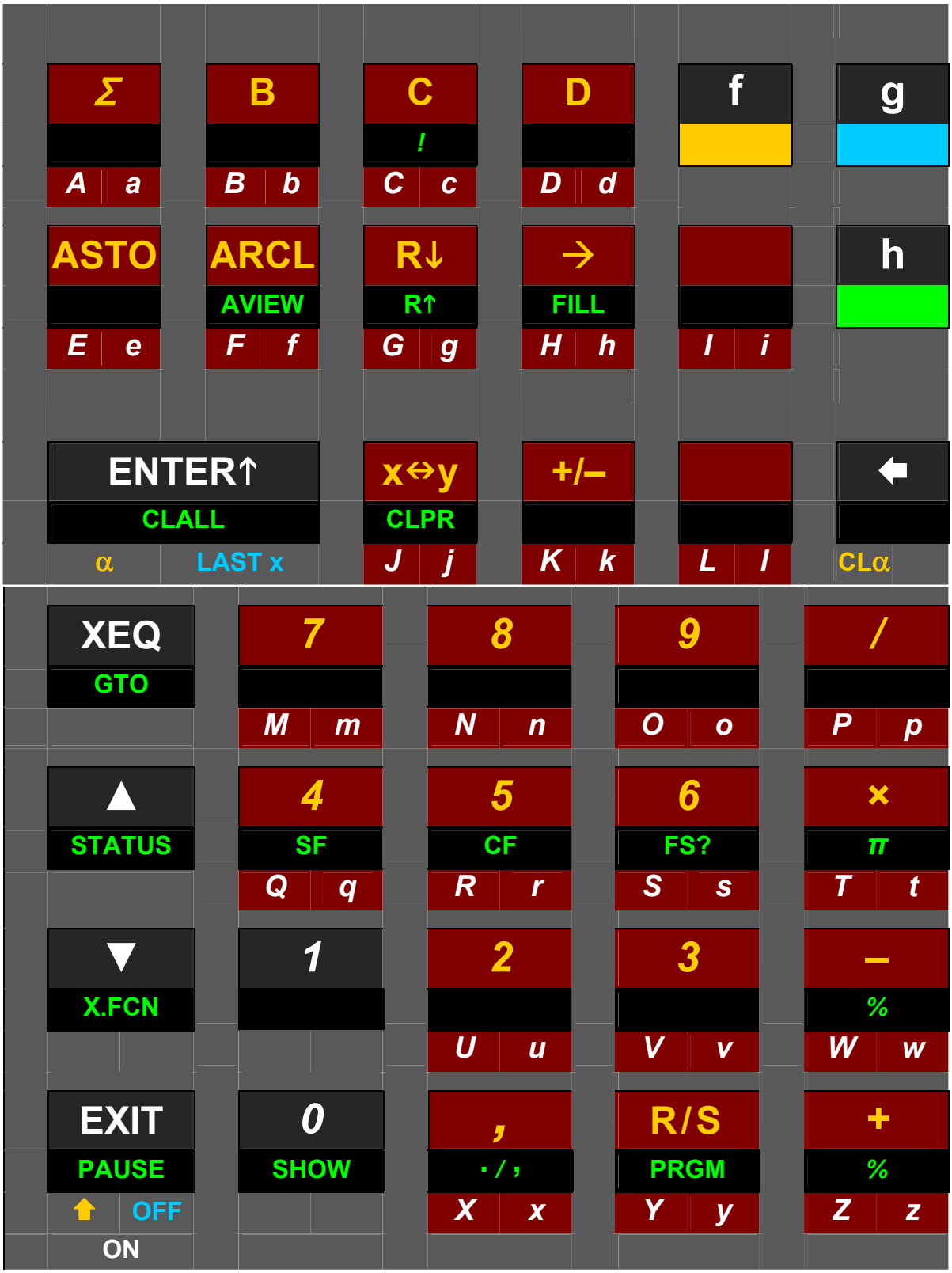


Keyboard layout:

- CPX may be combined with +, -, x, /, ±, x^2, √x, 1/x, //, !, Γ, π, |x|, RND, as well as e.g. (HYP) SIN, COS, TAN, logs 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 (except Σ+) will stay as they are in FLOAT.



Active operations in alpha mode. The alpha register is 15 bytes long. ➡ 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. PAUSE will insert a space.

ADDRESSING REGISTERS

1	User input	[x= ?] or any of the other comparisons			[RCL], [STO], [VIEW], [x≷], [DSE], [ISG], [DSZ], [ISZ], [CF] and the other flag commands, [FIX], [SCI], [ENG], [CB] and many more bit commands			
	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	closes alpha. **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 (+, –, ×, /) as well as MAX and MIN may precede step 2.

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

[3] Register and flag numbers may be [0] [0] … [9] [9], bit numbers [0] [1] … [6] [4], number of decimals [0] … [11] . For numbers <10, you may key in e.g. [5] [ENTER↑] instead of [0] [5] where applicable.

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.

Indicator	***INPUT*** **a**	**b**	**d**	**h**	**8**	***STO*** **PRG**
Set by operation	αON X.FCN	BIN	DEC	HEX	OCT	PRGON
Cleared by operation	αOFF ENTER	DEC FLOAT FRACT HEX OCT	BIN FLOAT FRACT HEX OCT	BIN DEC FLOAT FRACT OCT	BIN DEC FLOAT FRACT HEX	PRGOFF

Indicator	***360***	**g**	**HMS**	***RAD***	**/c**
Set by operation	DEG	GRAD	H.MS >H.MS TIMER	RAD ACOSH ASINH ATANH	FRACT, 2^{nd} [.] in input (\HMS)
Cleared by operation	GRAD RAD ACOSH ASINH ATANH	DEG RAD ACOSH ASINH ATANH	FLOAT >H	DEG GRAD	BIN DEC FLOAT HEX OCT

INPUT, STO, 360, and RAD are annunciators. Outside integer modes, everything else is indicated in the text line. An active timer application is signaled by T, the different date modes by D.MY or M.DY. Defaults Y.MD and FLOAT are not indicated.

Within integer modes, word size and complement setting are shown in the text line. The exponent is used for the further indications: its first digit shows the base, a “c” signals a carry bit set, an “o” an overflow.

Any inputs are interpreted according to the modes set at input time.

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 in the rightmost column, as are functions deviating from known ones carrying the same name. 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 in listings	Keys to press	Works in modes	Remarks
αAPP	h X.FCN **αAPP** *char*	All	Switches to alpha mode for the input of 1 character, appends this to the alpha register, and returns to the mode set before. Eventually, this equals the sequence AON *char* AOFF.
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	
αIP	h X.FCN **αIP**	\a	Like AIP in HP-42S.
ALL	h X.FCN **ALL**	DECM	
αLENG	h X.FCN **αLENG**	\a	Like ALENG in HP-42S.
AND	h AND	b, 8, d, h	
αOFF	f α	a	Toggles alpha mode like AOFF/AON in HP-42S. Not programmable. *) Please see the table for register addressing above for details.
	ENTER↑		
αON	f α	\a	
	ENTER↑	*)	
αRCL	f RCL *reg*	a	See opportunities for *reg* in the table above.
αROT	h X.FCN **αROT**	\a	Like AROT in HP-42S.
αSHIFT	h X.FCN **αSHIFT**	All	Like ASHF in HP-42S.
ASIN	g SIN^{-1}	DECM, H.MS	
ASINH	g HYP^{-1} SIN	DECM	

Name in listings	Keys to press	Works in modes	Remarks
ASR	h X.FCN **ASR** *n*	b, 8, d, h	Will work like *n* consecutive ASRs on HP-16C. See the opportunities for *n* in the table above.
αSTO	f STO *reg*	a	See opportunities for *reg* in the table above.
ATAN	g TAN^{-1}	DECM, H.MS	
ATANH	g HYP^{-1} TAN	DECM	
αVIEW	h SHOW	a	
	h VIEW		
	h X.FCN **αVIEW**	\a	
α→x	h X.FCN **α → x**	\a	Like ATOX in HP-42S.
BASE+	+	b, 8, d, h	
BASE–	–		
BASE×	×		
BASE/	/		
BASE+/–	+/–		
BC?	h X.FCN **BC?** *n*	b, 8, d, h	Tests the *n*-th bit in ***x***. See the opportunities for *n* in the table above.
BESTF	h X.FCN **BESTF**	DECM	Selects the best curve fit model according to the correlation found like in HP-42S.
BIN	f 2	\a	Sets binary integer mode.
BST	▲	PRG	Go 1 step back in program memory. Not programmable.
		\PRG	Go 1 step back without executing this step.
BS?	h X.FCN **BS?** *n*	b, 8, d, h	Works in analogy to “BC?”.
β(x,y)	h X.FCN **β(x,y)**	DECM	Calculates Euler’s Beta function.
B#	h X.FCN **B#**	b, 8, d, h	Counts bits set like **#B** on HP-16C.

Name in listings	Keys to press	Works in modes	Remarks
C …	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. The three dots will be replaced in the listing by the name of the function attached.
CB	h X.FCN **CB** *n*	b, 8, d, h	Clears the *n*-th bit in ***x***. See opportunities for *n* in the table above.
CEIL	h X.FCN **CEIL**	DECM	Computes largest integer ≤ ***x***.
CF	h CF *n*	All	See opportunities for *n* in the table above.
CLA	f CL α	All	Clears the alpha register.
CLALL	h CLALL	\PRG	Global clear after confirmation. Not programmable.
CLPR	h CLPR	PRG	Clears current program after confirmation. Not programmable.
CLRG	h X.FCN **CLRG**	All	Clears all general purpose registers.
CLΣ	h CLΣ	DECM	
CLSTK	0 h FILL	All	
CLX	h CLx	All	CPX h CLx clears **x** and **y**.
COMB	f Cx,y	DECM	
CONJ	h CONJ	DECM	Changes the sign of ***y***.
CONST	RCL C	DECM	Calls the catalog of constants like in HP35s. Not programmable. The constants contained are listed in a separate table below. ▲ and ▼ browse the catalog. Choose the constant displayed by pressing ENTER↑ . This will recall the constant into **x**. CPX RCL C will clear **y** in recalling.
CORR	g r	DECM	
COS	f COS	DECM, H.MS	
COSH	f HYP COS	DECM	

Name in listings	Keys to press	Works in modes	Remarks
DATE	h X.FCN **DATE**	DECM	Adds a number of days in ***x*** on a date in ***y*** and displays the resulting date including the day of week (Sunday = 7). This function works like in HP-12C.
DAY	h X.FCN **DAY**	DECM	Takes ***x*** as a date and returns an integer indicating the day of week (Sunday = 7) and displays the day of week in the dot matrix.
ΔDAYS	h X.FCN **ΔDAYS**	DECM	Calculates the number of days between 2 dates ***x*** and ***y***. Function works like in 12C.
DEC	f 10	\a	Sets integer decimal mode.
DEG	g DEG	DECM	
DENMAX	h X.FCN /c *n*	DECM	Sets the maximum denominator for fractions.
DISP	h X.FCN **DISP** *n*	DECM	Changes the number of decimals while keeping the mode (FIX, SCI, ENG). See opportunities for *n* in the table above.
DSE	f DSE *reg*	PRG	See opportunities for *reg* in the table above.
DSZ	h X.FCN **DSZ** *reg*		
D.MY	h X.FCN **D.MY**	DECM	Sets the format for date calculations.
Δ%	f Δ%	DECM	
D→J	h X.FCN **D→J**	DECM	Assumes **x** containing a date and converts it to a Julian day number.
D→R	h X.FCN **D→R**	DECM	Assumes **x** containing radians and converts them to degrees. Mode is kept constant.
EEX	E	DECM	
ENG	f ENG *n*	DECM	See opportunities for *n* in the table above.
ENTER	ENTER↑	All	
EXIT	EXIT	All	Exits X.FCN and any other menus or functions with pending input. Not programmable.
EXPF	h X.FCN **EXPF**	DECM	Selects the exponential curve fit model.
e^x	g e^x	DECM	
e^x -1	h X.FCN e^x-1	DECM	
FB	h X.FCN **FB** *n*	b, 8, d, h	Flips the *n*-th bit in ***x***. See the opportunities for *n* in the table above.

Name in listings	Keys to press	Works in modes	Remarks
FCSTx	h X.FCN $\hat{x}$	DECM	FCSTx (*FCSTy*) predicts x (*y*) for a given y (*x*) according to the curve fit model chosen. See L.R. for more.
FCSTy	f $\hat{y}$		
FC?	h X.FCN **FC?** *n*	All	See opportunities for *n* in the table above.
FC?C	h X.FCN **FC?C** *n*		
FF	h X.FCN **FF** *n*	All	Inverts ("flips") the flag specified. See the opportunities for *n* in the table above.
FILL	h FILL	All	Copies ***x*** in **y**, **z**, and **t**.
FIX	f FIX *n*	DECM	See opportunities for *n* in the table above.
FLOAT	f .d	\a	Works like DECM in HP-42S. Additionally, H.MS data in **x** will be converted to decimal.
FLOOR	h X.FCN **FLOOR**	DECM	Computes the smallest integer ≥ ***x***.
FP	g FP	DECM	
FRACT	g b/c	DECM	Sets fraction mode like in HP-32SII.
FS?	h FS? *n*	All	See opportunities for *n* in the table above.
FS?C	h X.FCN **FS?C** *n*		
GAMMA	h X.FCN **GAMMA**	DECM	
GRAD	g GRAD	DECM	
GTO	h GTO *label*	PRG	Like in HP32S.
		\PRG	
	h GTO . *label*	\PRG	
	h GTO . .		
HEX	g 16	\a	Sets hexadecimal integer mode.
H.MS	g H.MS	DECM	Sets H.MS mode.
H.MS+	+	H.MS	
H.MS–	–	H.MS	
INT	h INTEG *label*	DECM	Integration parameters will be transferred like in 15C.
IP	f IP	DECM	

Name in listings	Keys to press	Works in modes	Remarks
ISG	g ISG *reg*	PRG	See opportunities for *reg* in the table above.
ISZ	h X.FCN **ISZ** *reg*		
J→D	h X.FCN **J→D**	DECM	Assumes **x** containing a Julian day number and converts it to a date.
LASTX	g LASTx	All	CPX g LASTx recalls **x** and **y**.
LBL	f LBL *label*	PRG	
LEAP?	h X.FCN **LEAP?**	PRG	Checks if the integer part of ***x*** corresponds to a leap year.
LINF	h X.FCN **LINF**	DECM	Selects the linear curve fit model.
LJ	h X.FCN **LJ**	b, 8, d, h	
LN	f LN	DECM	
LNβ	h X.FCN **LNβ**	DECM	Calculates the natural logarithm of *β* or GAMMA, respectively. See these functions.
LNGAMM	h X.FCN **LNGAMM**		
LN1+	h X.FCN **LN1+**	DECM	
LOGF	h X.FCN **LOGF**	DECM	Selects the logarithmic curve fit model.
LOG$_Y$	f LGy	DECM	Calculates the logarithm for base ***y***.
LOG$_{10}$	f LOG	DECM	
LOG$_2$	f LG2	\a	Calculates the logarithm for base 2.
L.R.	h L.R.	DECM	Calculates the parameters of the fit curve (through the data points accumulated) according to the model selected. Returns A0 in **x** and A1 in **y**. In the linear model, A0 is the intercept and A1 the slope of the regression line.
MASKL	h X.FCN MASKL *n*	b, 8, d, h	Work like MASKL and MASKR on HP-16C, but with the parameter following the command instead of taken from **x**. See the opportunities for *n* in the table above.
MASKR	h X.FCN MASKR *n*		
MAX	h X.FCN **MAX**	\a	Returns the maximum of ***x*** and ***y***.
MEAN	f x̄	DECM	
MIN	h X.FCN **MIN**	\a	Returns the minimum of ***x*** and ***y***.

Name in listings	Keys to press	Works in modes	Remarks
MIRROR	h X.FCN **MIRROR**	b, 8, d, h	Reflects bit patterns (e.g. 000101 → 101000)
MOD	h MOD	\a	
M.DY	h X.FCN **M.DY**	DECM	Sets the format for date calculations.
n	h X.FCN **n**	DECM	Recalls the number of accumulated data points. Necessary for basic statistics.
NAND	h X.FCN **NAND**	b, 8, d, h	
NOP	h X.FCN **NOP**	PRG	
NOR	h X.FCN **NOR**	b, 8, d, h	
NOT	h NOT	b, 8, d, h	
OCT	g 8	\a	Sets octal integer mode.
OFF	g OFF	\PRG	Not programmable.
	h X.FCN **OFF**	PRG	
ON	ON	Calc. off	Not programmable.
	h X.FCN **ON**	PRG	
OR	h OR	b, 8, d, h	
π	h π	DECM	CPX h π puts π in **x** and clears **y** for using π in complex calculations.
	D	DECM	As long as no reassignment took place.
PAUSE	h PAUSE	PRG	Like PSE in HP42S.
PERM	g Py,x	DECM	
PRGOFF	h PRGM	PRG	Toggles programming mode. Not programmable.
PRGON		\PRG	
PROMPT	h X.FCN **PROMPT**	PRG	
PWRF	h X.FCN **PWRF**	DECM	Selects the power curve fit model.
Q(z)	f Qz	DECM	Like in HP-32E and HP-21S
RAD	g RAD	DECM	
RAND#	h RN#	\a	Like RAN in HP42S.

Name in listings	Keys to press	Works in modes	Remarks
RCL	RCL *reg*	\h, \a	See RCL+ for more details.
	f RCL *reg*	h	
RCLWS	h X.FCN **RCLWS**	b, 8, d, h	Recalls the word size set.
RCL+	RCL + *reg*	\h, \a (needs f in hex mode)	Calls the content of address *reg* , executes **OP** ***x*** on it and stores the result in **x**. RCL▲ (▼) takes the maximum (minimum) of the value in *reg* and ***x***. 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*		
RCL▲	RCL ▲ *reg*		
RCL▼	RCL ▼ *reg*		
RDX,	h ./,	DECM	Toggles radix mark.
RDX.			
RJ	h X.FCN **RJ**	b, 8, d, h	Works in analogy to LJ.
RL	h X.FCN **RL** *n*	b, 8, d, h	Will work like *n* consecutive RLs (RLCs) on HP-16C. See opportunities for *n* in the table above.
RLC	h X.FCN **RLC** *n*		
ROUND	g RND	DECM	Like RND in HP42S.
RR	h X.FCN **RR** *n*	b, 8, d, h	Will work like *n* consecutive RRs (RRCs) on HP-16C. See opportunities for *n* in the table above.
RRC	h X.FCN **RRC** *n*		
RTN	g RTN	PRG	
R/S	R/S	\PRG, \T	
		T	Starts/stops incrementing the timer.
R↑	h R↑	All	
R↓	R↓	All	
R→D	h X.FCN **R→D**	DECM	Assumes **x** containing degrees and converts them to radians. Mode is kept constant.
SB	h X.FCN **SB** *n*	b, 8, d, h	Sets the *n*-th bit in ***x***. See opportunities for *n* in the table above.
SCI	f SCI *n*	DECM	See opportunities for *n* in the table above.
SDEV	g s	DECM	

Name in listings	Keys to press	Works in modes	Remarks
SERR	h X.FCN **SERR**	DECM	Calculates $\frac{SDEV}{\sqrt{N}}$.
SEED	STO h RN#	DECM	
SF	h SF *n*	All	See opportunities for *n* in the table above.
SHOW	h SHOW	DECM	Shows the full mantissa.
		a	Shows the alpha register.
SIGMA	h X.FCN **SIGMA**	DECM	Calculates $SDEV \cdot \sqrt{\frac{N}{N-1}}$.
SIGNMT	h X.FCN **SIGNMT**	All	Sets sign-and-mantissa mode for integers.
SIN	f SIN	DECM, H.MS	
SINH	f HYP SIN	DECM	
SL	h X.FCN **SL** *n*	b, 8, d, h	Will work like *n* consecutive SLs on HP-16C. See the opportunities for *n* in the table above.
Σlnx	h X.FCN **Σlnx**	DECM	Recalls the respective statistical sum. These sums are necessary for the curve fitting models beyond pure linear. See below for more.
$\Sigma \ln^2 x$	h X.FCN **Σln²x**		
Σlny	h X.FCN **Σlny**		
$\Sigma \ln^2 y$	h X.FCN **Σln²y**		
SLV	h SOLVE *label*	DECM	
SR	h X.FCN **SR** *n*	b, 8, d, h	Will work like *n* consecutive SRs on HP-16C. See the opportunities for *n* in the table above.
SST	▼	PRG	Go 1 step forward in program memory. Not programmable.
		\PRG	Go 1 step forward executing this step.
STATUS	h STATUS	\PRG	Not programmable.
STO	STO *reg*	\h, \a	See STO+ for more details.
	f STO *reg*	h	
STOP	R/S	PRG	

Name in listings	Keys to press	Works in modes	Remarks
STO+	STO + *reg*	\h, \a (needs f in hex mode)	Calls the content of address *reg* , executes **OP *x*** on it and stores the result into said address. STO▲ (▼) takes the maximum (minimum) of the value in *reg* and ***x***. 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*		
STO▲	STO ▲ *reg*		
STO▼	STO ▼ *reg*		
SUM	RCL Σ+	DECM	
Σx	h X.FCN **Σx**	DECM	Recalls the respective statistical sum. These sums are necessary for basic statistics and linear curve fitting. Calling them by name greatly enhances readability of programs. These 13 statistical sums (in total, incl. *n*) shall be stored in registers 86 through 99.
Σx^2	h X.FCN **Σx^2**		
Σxy	h X.FCN **Σxy**		
Σy	h X.FCN **Σy**		
Σy^2	h X.FCN **Σy^2**		
Σ+	Σ+	DECM	
Σ–	h Σ−	DECM	
Σ(lnx·lny)	h X.FCN **Σ(lnx·lny)**	DECM	Recalls the respective statistical sum. Also these sums are necessary for the curve fitting models beyond pure linear.
Σ(x·lny)	h X.FCN **Σ(x·lny)**		
Σ(y·lnx)	h X.FCN **Σ(y·lnx)**		
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). See opportunities for *reg* in the table above. Clears the register specified.
TIMEX	h TIMER	T	Leaves the timer application (clears indicator T, but H.MS stays on).
UNSIGN	h X.FCN **UNSIGN**	All	
VIEW	h VIEW *reg*	All	See opportunities for *reg* in the table above.

Name in listings	Keys to press	Works in modes	Remarks
W	h X.FCN **W(x)**	DECM	Calculates Lambert's W for a given x ≥ -1/e
W $^{-1}$	h X.FCN **W** $^{-1}$	DECM	Inverts W, i.e. calculates x for a given W (≥ -1).
WMEAN	h X.FCN **WMEAN**	DECM	Calculates the weighted mean.
WSIZE	h X.FCN **WSIZE** *n*	All	Works like WSIZE on HP-16C, but with the parameter following the command instead of taken from **x**. See the opportunities for *n* in the table above.
XEQ	XEQ *label*	PRG	Calls the respective subroutine.
		\PRG	Executes the respective program.
XNOR	h X.FCN **XNOR**	b, 8, d, h	
XOR	h XOR	b, 8, d, h	
x ^{2}	g x^2	\a	
X.FCN	h X.FCN	DECM, \PRG	Calls the catalog of extra real functions and sets alpha mode to allow for keying in names of operations. Not programmable. ▲ and ▼ browse the catalog. Choose the operation displayed by pressing ENTER↑ . This will execute the function. – CPX h X.FCN calls the catalog of extra complex functions.
		b, 8, d, h, \PRG	Calls the catalog of extra integer functions. See above for more.
x!	h !	DECM	
x→α	h X.FCN **x → α**	\a	
x ↔	h x≷ *reg*	All	See opportunities for *reg* in the table above. CPX h x≷ will exchange ***x*** and ***y*** with the register specified and the next adjacent register.
x ↔***y***	x≷y	All	This performs **Re <> Im** if a complex operation was executed before. CPX x≷y will exchange ***x*** and ***y*** with ***z*** and ***t***.

Name in listings	Keys to press	Works in modes	Remarks
x< … ?	f x < ? *arg*	\a	Compares ***x*** with *arg*. See opportunities for *arg* in the table above. The three dots will be replaced in the listing by *arg* according to the samples given in said table.
x≤ … ?	g x ≤ ? *arg*		
x= … ?	f x = ? *arg*		
x≠ … ?	g x ≠ ? *arg*		
x≥ … ?	f x ≥ ? *arg*		
x> … ?	g x > ? *arg*		
Y.MD	h X.FCN **Y.MD**	DECM	Sets the format for date calculations.
y^x	g y^x	DECM	
	C	DECM	As long as no reassignment took place.
z(p)	g zP	DECM	Like in $\mathbf{Q^{-1}}$ in HP-32E and $\mathbf{z_P}$ in HP-21S
0, 1	0 , 1	All	Numeric input. The top left 6 keys will be used for input of hexadecimal numbers >10 in HEXM by default.
2 … 7	2 …7	\b	
8, 9	8 , 9	\b, \8	
A … F	A etc.	h	
[.] or [,]	.	DECM	Inserts the radix mark as selected.
		a	Inserts a point if RDX**.** , else a comma.
	h ./,	a	Inserts a comma if RDX**.** , else a point.
[ ] or [/]	.	/c	First input inserts a space, second a fraction mark, e.g. 2 . 3 . 4 results in 2 ¾ in the display.
[°]	.	H.MS	Separates degrees (or hours) from minutes and seconds.
[.]	.	Y.MD, D.MY, M.DY	Separates the leading unit in date modes. It is left to the user to decide if a number displayed represents a date or not.
1/x	f 1/x	DECM	
	B	DECM	As long as no reassignment took place.
1COMPL	h X.FCN **1COMPL**	All	Like 1's complement in HP-16C.
2COMPL	h X.FCN **2COMPL**	All	Like 2's complement in HP-16C.
2^x	g 2^x	\a	

Name in listings	Keys to press	Works in modes	Remarks
10^x	g 10^x	DECM	
+	+	DECM	
–	–		
×	×		
/	/		
+/–	+/–		
//	g //	DECM	Calculates $\left(\frac{1}{x}+\frac{1}{y}\right)^{-1}$.
√‾	f √x	\a	
%	g %	DECM	
%T	h X.FCN **%T**	DECM	Calculates $\frac{x}{y}\cdot 100$.
%+	h %+	DECM	Adds a markup of ***x*** % to ***y***.
%–	h %–	DECM	Subtracts a discount of ***x*** % from ***y***.
→BIN	→ f 2	\a	Shows ***x*** in binary or integer 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 angular mode and converts them to degrees or gon, respectively. Mode is kept constant.
→GRAD	→ g GRAD		
→H	→ f H	H.MS	Takes the hours or degrees in **x** and converts them into decimal numbers.
→HEX	→ g 16	\a	Works like →BIN, but hexadecimal.
→H.MS	→ g H.MS	DECM	Assumes **x** containing *decimal* hours or degrees and converts them in the format HHH.MMSS.
→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.

Name in listings	Keys to press	Works in modes	Remarks
→REC	f ▶R	DECM	Assumes **x** and **y** containing the coordinates r and ϑ and converts them to x and y.
	f ◀	b, \PRG	Shift the display window like in HP-16C. Not programmable.
	g ▶		

TABLE OF CONSTANTS

This lists all constants contained in the menu CONST. The constants π and **e** are found on the keyboard directly.

Name	Number	Dimension	Remarks
a_0	5,291772083E-11	m	Bohr radius $= \alpha/4\pi R_\infty$
α	7,297352533E-3	1	Fine-structure constant $= e^2/4\pi\varepsilon_0\hbar c$
atm	101325	Pa / atm	Standard pressure
c	299792458	m / s	Vacuum speed of light
c_1	374177107E-16	W m²	First radiation constant $= 2\pi \cdot h \cdot c^2$
c_2	0,014387752	m K	Second radiation constant $= hc/k$
ε_o	8,854187817E-12	$\frac{A \cdot s}{V \cdot m}$	Electric constant, vacuum permittivity $= 1/\mu_0 c^2$
eV	1,602176462E-19	J = A s V	= Electron charge x 1V
F	96485,3415	A s / mol	Faraday's constant
g	9,80665	m/s^2	Standard earth acceleration
G	6,6742867E-11	$\frac{m^3}{kg \cdot s^2}$	Newton's gravitation constant
G_o	7,748091696E-5	1 / Ω	Conductance quantum $= 2e^2/h$
γ_{EM}	0,5772156649015328606	1	Euler-Mascheroni's constant

Name	Number	Dimension	Remarks
γ_p	267522212	$\frac{1}{s \cdot T}$	Proton gyromagnetic ratio $= 2\mu_P / \hbar$
h	6,62606876E-34	J s	Planck constant
ħ	1,054571596E-34	J s	$= h/2\pi$
k	1,3806503E-23	J/K	Boltzmann constant $= R/N_A$
λ_c	2,426310215E-12	m	Compton wavelength $= h/m_e c$
λ_{cn}	1,319590898E-15	m	Compton wavelength of neutron $= h/m_n c$
λ_{cp}	1,321409847E-15	m	Compton wavelength of proton $= h/m_p c$
m_e	9,10938188E-31	kg	Electron mass
m_n	1,67492716E-27	kg	Neutron mass
m_p	1,67262158E-27	kg	Proton mass
m_μ	1,88353109E-28	kg	Myon mass
m_u	1,66053873E-27	kg	Atomic unit mass
μ_o	1,2566370614E-6	N/A^2	Magnetic constant, vacuum permeability $= 4\pi \cdot 10^{-7}$
μ_B	9,27400899E-24	J / T	Bohr's magneton $= e\hbar/2m_e$
μ_N	5,05078317E-27	J / T	Nuclear magneton $= e\hbar/2m_p$
μ_e	-9,28476362E-24	J / T	Electron magnetic moment
μ_μ	-4,49044813E-26	J / T	Myon magnetic moment
μ_n	-9,662364E-27	J / T	Neutron magnetic moment
μ_p	1,410606633E-26	J / T	Proton magnetic moment
N_A	6,02214199E23	1/mol	Avogadro's number
r_e	2,817940285E-15	m	Classical electron radius $= \alpha^2 \cdot a_0$
Φ	1,6180339887498948482	1	Golden ratio $\frac{1+\sqrt{5}}{2}$

Name	Number	Dimension	Remarks
Φ_o	2,067833636E-15	V s	Magnetic flux quantum $= h/2e$
R	8,314472	$\frac{J}{mol \cdot K}$	Molar gas constant
R_K	25812,80756	Ω	Von Klitzing constant
R_∞	10973731,5685	1 / m	Rydberg constant $= \alpha^2 m_e c/2h$
σ	5,6704E-8	$\frac{W}{m^2 \cdot K^4}$	Stefan Boltzmann constant
T_o	273,15	K	= 0°C, standard temperature
V_m	0,022413996	m^3/mol	Molar volume of ideal gas
Z_o	376,730313461	Ω	Characteristic impedance of vacuum $= \sqrt{\mu_0/\varepsilon_0} = \mu_0 c$
∞		1	Infinity

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

1. **TIMER** (accuracy down to 0.1s is sufficient, we can't press keys more precisely – included in the index already but not implemented yet)
2. **T**, **CHISQ**, and **F** distributions (in this order) and their inverses (not in the index yet)

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.
1.3	17.1.09	Added AIP, ALENG, ARCL, AROT, ASHF, ASTO, ATOX, XTOA, AVIEW, CLA, PROMPT (all taken from 42S), CAPP, FC?C, FS?C, SGMNT, and the …# commands; renamed NBITS to BITS and STOWS to WSIZE; specified the bit commands closer; deleted the 4 carry bit operations.
1.4	10.2.09	Added CONST and a table of constants provided, D>J and J>D, LEAP?, %T, RCL and STO ▲ and ▼, and 2 forgotten statistics registers; deleted CHS, EMGAM, GSB, REAL and ZETA; purged and renamed the bit operations; renamed many commands.