

HP 10bII+ RPN Mode

Manual addendum for the HP 10bII+ Financial Calculator — Prototype firmware EDED

This addendum covers only the RPN mode added by prototype firmware EDED. For everything else, see the HP 10bII+ Financial Calculator User's Guide.

Prototype firmware EDED. This addendum describes prototype firmware. Its behaviour may change in later releases.

Disclaimer. This is unofficial, experimental firmware and documentation. It is not made, supported or endorsed by HP Inc. or its licensees, and it is provided as is, without any warranty. Installing it is at your own risk and may void the manufacturer's warranty. Flashing can fail, and a failed or interrupted flash, low batteries or an incompatible unit can leave the calculator unusable. Back up your programs and data first, because installing firmware or changing settings can erase memory. Results have not been certified; check important calculations independently. Nobody involved accepts liability for damage, data loss or any consequence of using it. HP and HP 10bII+ are trademarks of HP Inc., used only to identify the compatible hardware.

About RPN mode

The standard HP 10bII+ has two ways of working: **Chain** mode and **Algebraic** mode. Firmware EDED adds a third one, **RPN** (Reverse Polish Notation), which works like the classic HP-12C and HP-15C. You key in the numbers first and then press the operation. RPN has no  step, no parentheses and no pending operations.

RPN mode uses the calculator's own arithmetic, financial and statistical routines. The only difference is the order in which you press the keys. **Answers in RPN mode are digit-for-digit identical to answers in Chain and Algebraic mode.** RPN mode also has two functions that Chain and Algebraic mode do not have: the common logarithm **LOG** and its inverse **10^x** (section 5).

How keys are shown

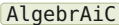
In this addendum	Means
	Press the key with INPUT printed on it.
	Press the orange shift key, then the key with SWAP printed in orange below it (here: K).
	Press the blue shift key, then the key with Alg/Chain printed in blue above it (here: ON).
12	Type the digits 1 and 2.

In this addendum	Means
17.00	What the display shows. A trailing  means you are still typing the number.

In RPN mode,  and  both work as the **ENTER** key found on other HP RPN calculators. This addendum writes it as .

1. Selecting RPN mode

Press   (blue shift, then ON). Each press moves to the next mode and briefly shows its name:

Mode before	Keys	New mode	Message
Chain	 	Algebraic	
Algebraic	 	RPN	
RPN	 	Chain	

- To see which mode is active without changing it, press   , as usual. In RPN mode this shows  and leaves the stack alone.
- The mode is remembered when you turn the calculator off and on again, like Chain and Algebraic. The whole RPN stack (X, Y, Z, T and LAST X) is kept too.
- When you switch **into** RPN mode:
 - the number in the display becomes X;
 - Y, Z, T and LAST X are cleared to zero;
 - any unfinished Chain or Algebraic calculation is dropped;
 - a stored **K** constant is cancelled.

2. The stack

RPN mode keeps four numbers, called the **stack**:

Register	Role
T	top
Z	
Y	the first number of a two-number operation
X	the number in the display; the second number of a two-number operation

- **Two-number operations** such as  combine Y and X. The result goes into X and the stack **drops**: Z moves into Y, and T moves into Z. T keeps its value,

so a number stored in T is **copied down** each time the stack drops. You can use this for repeated calculations with a constant (see example 8.8).

- **Stack lift.** When you start typing a new number after most operations, the stack **lifts** first: Z moves into T, Y moves into Z, and X moves into Y. The old T is lost. Some keys turn stack lift **off**. The next number you type then **replaces** X instead of pushing it up.

Key	What it does to the stack	Stack lift afterwards
Digits,  ,  	Start a new number. If lift is on, the stack lifts first.	on
 or  (ENTER)	Copies X into Y ($Y \rightarrow Z$, $Z \rightarrow T$; the old T is lost).	off : the next number overwrites the copy in X
 (CLx)	Sets X to 0. Y, Z and T are kept.	off
	While typing: deletes the last digit. Otherwise: CLx, like  .	unchanged while typing; off after CLx
	Changes the sign of the number being typed, or of X.	unchanged
 ,  	Add or remove a statistics pair: x is taken from X and y from Y (y  x  , see section 7). X shows n. Y keeps y; Z and T are kept. LAST X = x.	off : the next number replaces n
  n	Stores X. The stack is not changed.	unchanged
  n (also   )	$X + Rn$ ($X - Rn$, $X \times Rn$, $X \div Rn$) replaces X; Y, Z and T do not move (section 6).	on
 n,  (LAST X),   ,   , the sums (  ,   ,   ,   ,   ,  ),   ,  	Put a value into X. If lift is on, the stack lifts first.	on
Functions with two results (  ,   ,   ,   , ...)	Main result in X, second result in Y; the stack lifts (see section 7.1).	on

Key	What it does to the stack	Stack lift afterwards
 C ALL	The usual CLEAR ALL. In addition, clears X, Y, Z, T and LAST X.	on
All other operations and functions	See sections 4–7.	on

3. The keyboard in RPN mode

Most keys do exactly what the User's Guide says. These keys work differently in RPN mode:

Key	In RPN mode
INPUT	ENTER . Copies X into Y and turns stack lift off. It also stores X in the INPUT register for CFj and  AMORT , and the INPUT annunciator lights. The annunciator stays lit while you key in the next number (digits,  ,  E ,  ,  , RCL , K ,  STO), and goes out with the next operation. (For statistics, RPN mode takes x from X and y from Y: y INPUT x Σ+ , section 7.)
=	ENTER , the same as INPUT . Exception: while amortization results are shown, = steps through PRIN, INT and BAL (see example 8.11).
C	CLx . Clears X only; Y, Z and T are kept. It also clears messages, as usual.
RM	R↓ (roll down): $X \leftarrow Y$, $Y \leftarrow Z$, $Z \leftarrow T$, and $T \leftarrow X$.
 LOG (orange RM)	LOG : the common (base-10) logarithm of X (section 5).
 SWAP (orange K)	x↔y : exchanges X and Y. Straight after a function with two results, Y holds the second result, so  SWAP shows it, and the X/Y annunciator changes from X to Y (section 7.1).
K	LAST X : recalls the value X held before the last operation (see section 5). The K constant is not used in RPN mode.
→M	Stores X in the M register, as usual.
M+	Adds X to the M register, as usual.
RCL RM	Recalls the M register into X (the stack lifts). RM alone is R↓ in RPN mode.
 10^x (orange M+)	10^x : 10 to the power X (section 5).

Key	In RPN mode
blue HYP (blue RM), blue INV (blue M+)	As usual.

RPN mode has no parentheses and no K constant: the stack does their job, so orange **()** and orange **10^x** are used for LOG and 10^x. To roll the stack the other way (R↑), press **RM** three times. You use the M register with **→M**, **M+** and **RCL** **RM**. The 20 numbered registers are covered in section 6.

4. Arithmetic and two-number functions

Key in the first number, press **INPUT**, key in the second number, and press the operation. The first number is in Y and the second in X.

Keys	Result in X	Stack afterwards	LAST X
+ - × ÷	$Y + X$, $Y - X$, $Y \times X$, $Y \div X$	drops	old X
orange y^x	Y to the power X	drops	old X
%	$Y \times X \div 100$	Y is kept (like the HP-12C), so a following + or - adds the percentage to Y or subtracts it	old X
orange %CHG	percent change from Y to X: $100 \times (X - Y) \div Y$	drops	old X
blue nCr , blue nPr	combinations or permutations of Y items taken X at a time	drops	old X
blue df,t2P	Student's t: the lower-tail probability for Y degrees of freedom and the t value in X. With blue INV first, X is a probability and the result is the t value (section 5)	drops	old X
orange DATE	the date that is X days after the date in Y	drops	old X
orange ΔDAYS	the number of days from the date in Y to the date in X	drops	old X

Because each operation acts straight away, you never need **=** or parentheses. Intermediate results wait on the stack until you need them (see example 8.3).

Errors. If a two-number operation cannot be done, for example 1 **INPUT** 0 **+**:

- the usual error message appears (here **Error / 0**). If a number is outside what the function accepts, for example degrees of freedom that are not a positive whole number for **blue** **df,t&P**, a probability that is not between 0 and 1 for **blue** **INV** **blue** **df,t&P**, or more items taken than there are for **blue** **nCr** or **blue** **nPr**, the ERROR and FUNC annunciators light and the display keeps showing X. If the answer would be too large (or infinite, as for a probability of exactly 0 or 1), the display shows **0fLo**;
- the stack is **not changed**;
- stack lift is turned off, so the next number you type **replaces** the bad X;
- LAST X is not changed.

5. One-number functions and LAST X

One-number functions work on X and put the result in X. The rest of the stack is not moved, and stack lift is turned on afterwards. These functions include

orange **1/x**, **orange** **√x**, **orange** **x²**, **orange** **LN**, **orange** **e^x**, LOG (**orange** **()**), 10^x (**orange** **()**), **orange** **n!**, **orange** **RND**, **blue** **SIN**, **blue** **COS** and **blue** **TAN**, also with **blue** **INV** and **blue** **HYP**, and the normal-probability function **blue** **Z&P**.

LOG and 10^x. Press **orange** **()** (orange RM) for the common logarithm of X, and **orange** **()** (orange M+) for 10 to the power X. They work like **orange** **LN** and **orange** **e^x**: the result replaces X, Y, Z and T are kept, LAST X holds the number you started with, and stack lift is on afterwards (example 8.19).

- Exact powers of ten give exact answers: 1000 LOG is 3, 1 **orange** **E** 6 LOG is 6, and 2 10^x is 100. 10^x of any whole number from -499 to 499 is exact. Other results are correct to within one unit in the 12th digit.
- LOG of 0 or of a negative number is an error, shown and handled the same way as **orange** **LN** of such a number (see below).
- If 10^x is too large (X of 500 or more), the display shows **0fLo**, as for **orange** **e^x**. If it is too small (X below -499), the result is 0.
- LOG and 10^x are available only in RPN mode. In Chain and Algebraic mode, the same keys are the parentheses.

Random numbers. **blue** **RAND** lifts the stack like a recall (as RAN# on the HP-15C). Pressing it twice leaves two random numbers in X and Y. The seed is stored with **orange** **STO** **blue** **RAND**, as described in the User's Guide (example 8.17).

LAST X

Just before any operation uses up X, the calculator saves X as **LAST X**. Press **(K)** to recall it. This lets you reuse a number or undo a mistake (examples 8.6 and 8.7).

Saves X as LAST X	Leaves LAST X unchanged
$\boxed{+}$ $\boxed{-}$ $\boxed{\times}$ $\boxed{\div}$ $\boxed{y^x}$ $\boxed{\%}$ $\boxed{\%CHG}$ $\boxed{nCr}$ $\boxed{nPr}$ $\boxed{df,t\pm P}$ $\boxed{DATE}$ $\boxed{ADAYS}$, all one-number functions ($1/x$, $\sqrt{x}$, x^2 , LN , e^x , LOG , 10^x , $n!$, RND , the trigonometric functions, ...), the two-result functions that use X: $\boxed{X,r}$, $\boxed{Y,m}$, $\boxed{SL}$, $\boxed{SOYD}$, $\boxed{DB}$, and $\boxed{\Sigma+}$ $\boxed{\Sigma-}$ (they save x), and RCL arithmetic ($\boxed{RCL}$ $\boxed{+}$ n, ...)	$\boxed{INPUT}$ $\boxed{=}$, $\boxed{SWAP}$, $\boxed{RM}$, $\boxed{+/-}$, $\boxed{C}$, $\boxed{-}$, $\boxed{RAND}$, the sums (n , Σx , Σy , Σx^2 , Σy^2 , Σxy), $\boxed{ACCRU}$ and digits, $\boxed{IRR/YR}$, $\boxed{STO}$, $\boxed{RCL}$, $\boxed{\pi}$, $\boxed{K}$ itself, and the two-result functions that do not use X: $\boxed{X,y}$, $\boxed{Sx,Sy}$, $\boxed{\sigma x,\sigma y}$, $\boxed{\bar{x}w,b}$, $\boxed{NPV}$

- $\boxed{K}$ behaves like a recall. If stack lift is on, the stack lifts first. Straight after $\boxed{INPUT}$, LAST X overwrites the copy in X, as on the HP-12C and HP-15C.
- LAST X is cleared by $\boxed{C ALL}$ and when you switch into RPN mode.
- A two-number operation that ends in an error does **not** change LAST X.

If a one-number function cannot be done (for example $\boxed{\sqrt{x}}$ of a negative number), the calculator reports the error the same way as in Algebraic mode:

- X keeps its value, and LAST X holds that value;
- press $\boxed{C}$ to clear the error, then key in the number you meant: it takes the place of the bad one;
- if you type a new number without pressing $\boxed{C}$ first, the bad value is kept and lifted into Y.

Normal and Student's t probabilities

Both functions give the **lower tail**: the probability of a value below the one you enter. Press $\boxed{INV}$ first to go the other way, from a probability to the value. See example 8.18.

- **Normal distribution**, $\boxed{Z\pm P}$: a one-number function. Key in z and press $\boxed{Z\pm P}$ to get the probability for the standard normal distribution (mean 0, standard deviation 1). Key in a probability and press $\boxed{INV}$ $\boxed{Z\pm P}$ to get z. A probability that is not between 0 and 1 is an error, handled like any other one-number error (see above).
- **Student's t**, $\boxed{df,t\pm P}$: a two-number function. Key in the degrees of freedom, $\boxed{INPUT}$, the t value, and press $\boxed{df,t\pm P}$: the probability replaces both numbers and the stack drops. For the t value, key in the degrees of freedom, $\boxed{INPUT}$, the probability, and press $\boxed{INV}$ $\boxed{df,t\pm P}$. The degrees of freedom must be a positive whole number.
- **Upper tail**: subtract the result from 1 with 1 $\boxed{SWAP}$ $\boxed{-}$. For both tails together (a two-sided test), multiply the upper tail by 2.

- **A normal distribution with mean μ and standard deviation σ :** turn x into z first: x **INPUT** μ **-** σ **+** **blue** **Z $\rightarrow$ P**. To go from a probability back to x, use **blue** **INV** **blue** **Z $\rightarrow$ P**, then σ **x** μ **+**.
- As with the other functions, the answers are the same as in Chain and Algebraic mode.

6. Storage registers (STO and RCL)

The 20 numbered registers work as described in the User's Guide:

- **orange** **STO** 0-9 and **orange** **STO** **.** 0-9 store X in registers 0-19. **RCL** recalls them the same way.
- Storage arithmetic works: **orange** **STO** **+** n, **orange** **STO** **-** n, **orange** **STO** **x** n and **orange** **STO** **÷** n change register n by X.
- STO and storage arithmetic leave the stack, stack lift and LAST X unchanged.
- **RCL** lifts the stack (if lift is on) and puts the register's value in X. After **INPUT**, the recalled value overwrites the copy in X.
- **RCL** also recalls the financial registers (**RCL** **N**, **RCL** **PV**, **RCL** **orange** **P/YR**, ...) and M (**RCL** **RM**), with the same stack lift.
- **Register arithmetic on X (RCL arithmetic):** **RCL** **+** n, **RCL** **-** n, **RCL** **x** n and **RCL** **÷** n (n = 0-9 or **.** 0-9) work out $X + R_n$, $X - R_n$, $X \times R_n$ or $X \div R_n$ and put the answer in X. They work like the HP-15C. The register keeps its value, and Y, Z and T don't move. LAST X holds the old X, and the next number you type lifts the stack. RCL arithmetic also works with the registers **RCL** can recall directly, for example **RCL** **+** **PV**.
- If the answer can't be worked out (for example R_n is 0 for **÷**), the usual error message appears and nothing changes. Press **C** to see X again.
- To cancel **RCL** **+** before you press the register, press **C** or **←**. **ON** does not cancel it.
- RCL arithmetic works only in RPN mode. In Chain and Algebraic mode **RCL** **+** works as described in the User's Guide.

The PEND annunciator shows a waiting STO or RCL. In RPN mode, PEND lights as soon as you press **RCL** or **orange** **STO**, and it stays lit while the calculator waits for the register: after **RCL** **+** (or **-**, **x**, **÷**), after **orange** **STO** **+** (or **-**, **x**, **÷**), after **.** for registers .0-.9, and after the orange shift key for a shifted register. PEND goes out when you press the register, when you cancel with **C** or **←** (or with a key that is not a register), and when an error message appears.

- **ON** does not cancel a waiting STO or RCL, so PEND stays lit.
- If PEND is lit, the first **orange** **OFF** cancels the waiting STO or RCL and PEND goes out; press it again to turn the calculator off.

- The automatic power-off also cancels it, so PEND is never lit when you turn the calculator back on.
- Because RPN mode has no pending operations, PEND has no other meaning in RPN mode. In Chain and Algebraic mode, PEND means a pending operation, as described in the User's Guide.

See example 8.10.

7. Financial, statistics, dates, bonds and break-even

These keys work on X and use the calculator's own routines, so they give the same answers as in Chain and Algebraic mode.

- **Storing a value** (TVM, cash flows, bonds, break-even, business percentages): key in the number and press the key, as usual. The value stays in X, and the next number you type lifts it into Y. So the stack fills up with values you have already stored. This is harmless.
- **Calculating a result** (press the key without typing a number first): a single result **replaces X**; the stack does not lift. The exceptions are IRR/YR, ACCRU and the statistics sums, which lift the stack like a recall. The functions that give **two** results are different: they lift the stack and put both results on it (section 7.1).
- **Working a value out first:** you can calculate on the stack and then store the result, for example 14500 INPUT 1500 - PV (example 8.11).
- **TVM and amortization** (N I/YR PV PMT FV, P/YR, AMORT, ...) work as described in the User's Guide. For amortization, key in first INPUT last AMORT, then press = to step through PRIN, INT and BAL, as usual. Press AMORT straight after the second number: if INPUT is not lit, AMORT uses only X (periods 1 to X), as in Chain and Algebraic mode without INPUT.
- **Cash flows** (CFj, Nj, IRR/YR): as described in the User's Guide. To enter a cash flow with a count, key in the cash flow, INPUT, the count, CFj (INPUT is lit). A cash flow keyed in alone, with INPUT dark, is stored once. IRR/YR lifts the stack like a recall. NPV gives two results, NPV and NFV (section 7.1, example 8.14).
- **Statistics (HP-12C/15C order):** enter each pair as y INPUT x Σ+, that is, y first and x last. Σ+ (and Σ-, which removes a pair) takes **x from X and y from Y**. In all statistics results, "x" means the values taken from X and "y" means the values taken from Y: $\bar{x}, \bar{y}$, S_x, S_y , σ_x, σ_y , $\hat{x}, r$, $\hat{y}, m$, $\bar{x}w, b$ and the sums Σx , Σy , Σx^2 , Σy^2 , Σxy . After Σ+ or Σ-:
 - X shows n, the number of pairs;
 - Y still holds y, and Z and T are kept;

- LAST X holds x;
- stack lift is off, so the next number you type replaces n.
- **Correcting a pair:** enter the same pair again with Σ- (y INPUT x Σ-). Straight after the wrong Σ+, K brings the wrong x back. The sums (n, Σx , Σy , Σx^2 , Σy^2 , Σxy) work like a recall: if stack lift is on, the stack lifts first. Straight after Σ+ (lift off), a sum replaces n. Means, standard deviations, estimates and $\bar{x}_w$ give two results (section 7.1, example 8.12).
- **Statistics with single values (x only):** key in each value and press Σ+. As on the HP-15C, the number in Y is recorded as its y value. Clear Y first with C INPUT (or C ALL). Because Σ+ keeps Y and the next value replaces n, Y then stays 0 for every value. The x results (n, Σx , $\bar{x}$, S_x , σ_x) are always correct. They are identical to the results in Chain and Algebraic mode. If Y held some other number, that number is recorded as y for every value. The y results then describe that number, not your data (example 8.12).
- Statistics data are shared by all three modes. Pairs entered in RPN mode are also correct in Chain mode, and pairs entered in Chain mode (x INPUT y Σ+) are also correct in RPN mode.
- **Dates:** DATE and ADAYS take their two numbers from Y and X, like the two-number functions in section 4. For example, date INPUT days DATE. The date format keys (D.MY/M.DY, 360/Act) work as usual.
- **Bonds** (SetDate, MatDate, CPN%, YTM, PRICE, ...) and **break-even** (FC, VC, SP, PROFIT, UNITS): as described in the User's Guide. ACCRU lifts the stack like a recall, so PRICE ACCRU + gives the price including accrued interest.

7.1 Functions with two results

Some functions give two results. In Chain and Algebraic mode you see the second one by pressing SWAP. In RPN mode, both results go onto the stack: the **main result in X** and the **second result in Y**, as on the HP-12C (for example $\bar{x}$ and $\bar{y}$). SWAP ($x \geq y$) then shows the second result, and the X/Y annunciator changes from X to Y. Both results can be used straight away in a calculation.

Keys	Uses X?	X (displayed)	Y
Σ $\bar{x}, \bar{y}$	no	mean of x	mean of y
Σ S_x, S_y	no	sample standard deviation of x	sample standard deviation of y
Σ σ_x, σ_y	no	population standard deviation of x	population standard deviation of y
Σ $\bar{x}_w, b$	no	mean of x weighted by y	the b coefficient (y-intercept)

Keys	Uses X?	X (displayed)	Y
orange NPV	no	net present value	net future value
y orange $\hat{x},r$	yes: y	estimated x for that y	correlation coefficient r
x orange $\hat{y},m$	yes: x	estimated y for that x	slope m
year blue SL, blue SOYD, blue DB	yes: the year	depreciation for that year	remaining depreciable value

What happens to the rest of the stack:

- **Functions that do not use X** ($\hat{x},\hat{y}$, S_x,S_y , σ_x,σ_y , $\bar{x}w,b$, NPV) work like two recalls. If stack lift is on, the old X moves to Z and the old Y moves to T. If stack lift is off (straight after INPUT, C or $\Sigma+$), the old X is overwritten: the old Y moves to Z and the old Z moves to T. LAST X is not changed. Straight after $\Sigma+$, $\hat{x}$ therefore replaces n, as on the HP-12C.
- **Functions that use X** ($\hat{x},r$, $\hat{y},m$, SL, SOYD, DB) replace X with the main result, as a one-number function does, and push the second result in above it: the old Y moves to Z and the old Z moves to T. The old T is lost. LAST X holds the number you gave the function (the y, the x or the year).
- Stack lift is on afterwards.
- If the function cannot be done (for example no statistics data), the usual error message appears and the stack and LAST X are not changed.

Before	After a function that does not use X (lift on)	After a function that does not use X (lift off)	After a function that uses X
T = t, Z = z, Y = y, X = x	T = y, Z = x, Y = second, X = main	T = z, Z = y, Y = second, X = main	T = z, Z = y, Y = second, X = main; LAST X = x

Amortization. orange AMORT has three results (principal, interest and balance), so they are not put on the stack. Press $\equiv$ to step through them (example 8.11). The stack is not touched while you step.

8. Worked examples

For each example, start in RPN mode. The display shows two decimal places, which is the factory setting.

8.1 Simple addition: 2 + 3

Keys	Display	What happens
2 INPUT	2.00	Keys in 2 and copies it into Y.
3	3_	The new number goes into X; 2 waits in Y.
+	5.00	Y + X. The stack drops.

8.2 $3 \times 4 + 5$

Keys	Display	What happens
3 INPUT	3.00	Enters the first number.
4 x	12.00	3×4 .
5 +	17.00	$12 + 5$. No INPUT needed: the 12 is lifted into Y automatically.

8.3 A calculation with brackets: $(12 + 3) \times (7 - 2)$

No parentheses are needed. Work each bracket out, and the intermediate result waits in the stack.

Keys	Display	What happens
12 INPUT	12.00	
3 +	15.00	First bracket: $12 + 3$.
7 INPUT	7.00	Start the second bracket; 15 moves up to Z.
2 -	5.00	$7 - 2$. The 15 drops back into Y.
x	75.00	15×5 .

8.4 Percentages: adding 15 % to 200

Keys	Display	What happens
200 INPUT	200.00	The base amount.
15 %	30.00	15 % of 200. The 200 stays in Y.
+	230.00	$200 + 30$: the amount plus 15 %.

To take 15 % **off** instead, press **-** in place of **+**:

Keys	Display	What happens
200 INPUT 15 %	30.00	15 % of 200; 200 stays in Y.
-	170.00	$200 - 30$: the amount less 15 %.

8.5 Percent change

Keys	Display	What happens
100 INPUT	100.00	Old value.
125 orange %CHG	25.00	Change from 100 to 125 is +25 %.
55 INPUT 32 orange %CHG	-41.82	Change from 55 to 32 is -41.82 %.

8.6 Reusing a number with LAST X: 12×4 , then $\div 4$ again

Keys	Display	What happens
12 INPUT	12.00	
4 x	48.00	12×4 . LAST X holds 4.
K	4.00	LAST X brings the 4 back (48 is lifted into Y).
÷	12.00	$48 \div 4$: back to the original 12.

8.7 Undoing a mistake with LAST X

You meant $2 - 3$ but pressed **+**:

Keys	Display	What happens
2 INPUT 3 +	5.00	Oops: you wanted to subtract.
K -	2.00	LAST X (3) - gives back the 2.
3 -	-1.00	Do the subtraction you meant.

8.8 Using T as a constant: 100 growing at 5 % a year

Keys	Display	What happens
1.05 INPUT INPUT INPUT	1.05	Fills Y, Z and T with 1.05.
100 x	105.00	After year 1.
x	110.25	After year 2 (T is copied down each time).
x	115.76	After year 3.

8.9 Rolling and exchanging the stack

Keys	Display	What happens
1 INPUT 2 INPUT 3 INPUT 4	4	Stack: T=1, Z=2, Y=3, X=4.
RM	3.00	R↓: X=3, Y=2, Z=1, T=4.
RM	2.00	R↓ again.
RM	1.00	
RM	4.00	Four R↓ bring the stack back to where it started.
RM RM RM	1.00	Three R↓ roll the stack up by one place (R↑): X=1, Y=4, Z=3, T=2.
orange SWAP	4.00	x≥y: X=4, Y=1.

A common use of x≥y is fixing the order of a subtraction or division:

Keys	Display	What happens
7 INPUT 8	8	Oops: the numbers are the wrong way round for 8 – 7.
orange SWAP	7.00	x≥y: Y=8, X=7.
-	1.00	8 – 7.

8.10 Storage registers and storage arithmetic

Keys	Display	What happens
519 orange STO 2	519.00	Stores 519 in register 2.
C	0.00	Clears X.
RCL 2	519.00	Recalls register 2.
1.25 orange STO . 5	1.25	Stores 1.25 in register 15.
3 orange STO + . 5	3.00	Adds 3 to register 15.
C	0.00	
RCL . 5	4.25	Register 15 holds 4.25.

RCL arithmetic: multiply 44 by register 1 without disturbing the stack. Watch PEND while the calculator waits for the register.

Keys	Display	What happens
5 orange STO	5.00 PEND	The calculator waits for the register.
1	5.00	Stores 5 in register 1. PEND goes out.
44 RCL X	44 PEND	Waits for the register.
1	220.00	44 × 5. Register 1 still holds 5.
K	44.00	LAST X gives back the old X.

8.11 TVM: a car loan

A car costs 14,500 with 1,500 down. The loan is over 36 months at 10.5 % a year. What is the monthly payment? (End-of-period payments, the factory setting.)

Keys	Display	What happens
12 orange P/YR	12.00	12 payments per year.
36 N	36.00	36 monthly payments.
10.5 I/YR	10.50	10.5 % annual interest.
14500 INPUT 1500 -	13,000.00	Price less down payment, worked out on the stack.
PV	13,000.00	Stores the loan amount in PV.
0 FV	0.00	Nothing left at the end.
PMT	-422.53	The monthly payment (money paid out, so negative).

Continue with the amortization of the first year. **=** steps through the results:

Keys	Display	What happens
1 INPUT 12 orange AMORT	1-12	Amortize payments 1 to 12 (the first year).
=	-3,889.01	Principal paid in year 1.
=	-1,181.35	Interest paid in year 1.
=	9,110.99	Balance after 12 payments.

8.12 Statistics

Enter the pairs $(x, y) = (2, 3), (4, 5)$ and $(9, 10)$, y first and x last, as on the HP-12C and HP-15C. Then find the means of x and y , their average, the sum of x , and the estimated y and the slope for $x = 6$.

Keys	Display	What happens
 C STAT	StAt cLr	Clears the statistics memory.
C	0.00	Clears the message and X.
3 INPUT 2 Σ+	1.00	First pair: $y = 3$, then $x = 2$. The display shows n . Y still holds 3, and LAST X is 2.
5 INPUT 4 Σ+	2.00	Second pair ($x = 4, y = 5$). The 5 replaced n , because $\Sigma+$ turned stack lift off.
10 INPUT 9 Σ+	3.00	Third pair ($x = 9, y = 10$).
 $\bar{x}, \bar{y}$	5.00	Mean of x in X; mean of y (6) in Y. The mean replaces n , because $\Sigma+$ turned stack lift off.
 SWAP	6.00	$x \geq y$ shows the mean of y . The annunciator changes from X to Y.
+ 2 ÷	5.50	Both means are on the stack, so you can use them straight away: $(6 + 5) \div 2$.
 Σx	15.00	Sum of x ($2 + 4 + 9$). Like a recall, it lifts the stack: 5.50 moves up into Y.
6  $\hat{y}, m$	7.00	Estimated y for $x = 6$ in X; the slope m (1) in Y. The 15 moves up to Z and the 5.50 to T. LAST X is 6.
 SWAP	1.00	Shows the slope.

Single values. Find the mean and the standard deviation of 2, 4, 9 and 15. The calculator holds the last example's results, so Y is cleared first:

Keys	Display	What happens
 C STAT	StAt cLr	Clears the statistics memory.
C INPUT	0.00	Puts 0 into X and Y, so every value is recorded with $y = 0$.
2 Σ+	1.00	First value. The display shows n ; Y is still 0.
4 Σ+	2.00	Each new value replaces n (stack lift is off), so Y stays 0.
9 Σ+	3.00	
15 Σ+	4.00	

Keys	Display	What happens
orange $\bar{x}, \bar{y}$	7.50	Mean of the four values. With only x values there is no second result, so Y is not changed.
orange Sx, Sy	5.80	Sample standard deviation of the values.

8.13 Depreciation: straight-line

The example from the User's Guide: an asset costs 10,000, has a salvage value of 500 and a useful life of 5 years.

Keys	Display	What happens
10000 PV	10,000.00	Depreciable cost.
500 FV	500.00	Salvage value.
5 N	5.00	Useful life in years.
1 blue SL	1,900.00	Depreciation in year 1 in X; the remaining depreciable value (7,600) in Y. LAST X is 1.
orange SWAP	7,600.00	Remaining depreciable value after year 1.
2 blue SL	1,900.00	Year 2: 1,900 in X and 5,700 in Y. The year-1 values move up to Z and T.
orange SWAP	5,700.00	Remaining depreciable value after year 2.

8.14 Net present value and net future value

An investment of 1,000 returns 500 after one year and 600 after two years. Is it worth it at 10 % a year? (While you enter cash flows, **CF 0**, **CF 1**, ... appears briefly, then the amount.)

Keys	Display	What happens
1 orange P/YR	1.00	One cash flow per year.
10 I/YR	10.00	10 % a year.
1000 +/- CFj	CF 0	Initial investment (CF0).
500 CFj	CF 1	Year 1.
600 CFj	CF 2	Year 2.
orange NPV	-49.59	Net present value in X; net future value (-60.00) in Y.
orange SWAP	-60.00	Net future value.

The NPV is negative, so the investment earns less than 10 % a year.

8.15 Days between two dates

The factory date format is M.DY.

Keys	Display	What happens
3.012026 INPUT	3.01	1 March 2026 (M.DY format).
9.292026 orange ADAYS	212.00	Days from 1 March to 29 September 2026.

8.16 An error, and carrying on

Keys	Display	What happens
1 INPUT 0 ÷	Error / 0	Division by zero. The stack is not changed.
4 ÷	0.25	The 4 you type replaces the bad 0; $1 \div 4$.

8.17 Random numbers

Store a seed first, so that you get the same numbers as this example. With another seed, or none, the numbers are different.

Keys	Display	What happens
42 orange STO	42.00	Stores 42 as the seed, so you get the same numbers as below.
blue RAND	0.20	A random number between 0 and 1. Like a recall, it lifts the stack: 42 moves up into Y.
blue RAND	0.86	A second random number. The first one (0.20) moves up into Y.
+	1.06	Adds the two random numbers.

8.18 Normal and Student's t probabilities

The standard normal distribution: lower tail, upper tail and the inverse.

Keys	Display	What happens
orange DISP 4	0.0000	Shows four decimal places for these examples.
1.96 blue Z&P	0.9750	Probability that a standard normal value is below 1.96 (lower tail). LAST X is 1.96.
1 orange SWAP -	0.02500	$1 - 0.9750$: the upper tail, the probability of a value above 1.96.
.95 blue INV blue Z&P	1.6449	The z value with 95 % of the distribution below it.

A normal distribution with mean 100 and standard deviation 15. What share of values is below 130, and below which value are 95 % of them?

Keys	Display	What happens
130 INPUT 100 [-]	30.0000	$x - \mu$, with $x = 130$ and the mean $\mu = 100$.
15 [÷]	2.0000	Divided by the standard deviation $\sigma = 15$: $z = 2$.
blue ZaP	0.9772	Probability of a value below 130.
.95 blue INV	1.6449	The z value for 95 %.
blue ZaP		
15 [x] 100 [+]	124.6728	$z \times \sigma + \mu$: 95 % of values are below 124.67.

Student's t with 10 degrees of freedom.

Keys	Display	What happens
10 INPUT 2.228 blue df,tzP	0.9750	Degrees of freedom in Y, t in X: the lower-tail probability. The stack drops and LAST X is 2.228.
1 orange SWAP [-]	0.02501	The upper tail.
2 [x]	0.05001	Both tails together (two-sided).
10 INPUT .975 blue INV blue df,tzP	2.2281	The t value with 97.5 % below it, for 10 degrees of freedom.
orange DISP 2	2.23	Back to two decimal places.

8.19 Logarithms and powers of ten

The common logarithm of 1000, a power of ten, and the pH of a solution with a hydrogen-ion concentration of 0.00025 ($\text{pH} = -\text{LOG}$ of the concentration). Then back from pH 3.6 to the concentration.

Keys	Display	What happens
1000 orange [log]	3.00	$\text{LOG } 1000 = 3$, exactly. LAST X is 1000.
2 orange [10^x]	100.00	$10^2 = 100$, exactly. The 3 moves up into Y.
.00025 orange [log]	-3.60	LOG of the concentration.
[+/-]	3.60	$\text{pH} = 3.60$.
3.6 [+/-] orange [10^x]	0.00025	10 to the power -3.6 : back to the concentration (0.000251).

9. RPN compared with Chain and Algebraic

	Chain / Algebraic	RPN
2 + 3	2 [+] 3 [=]	2 INPUT 3 [+]

	Chain / Algebraic	RPN
Order of operations	Chain: left to right. Algebraic: $\times \div$ before $+$ $-$. Parentheses and pending operations.	None needed: each operation acts at once on Y and X.
INPUT	Separates two numbers; INPUT register	ENTER (also loads the INPUT register)
=	Completes the calculation	ENTER (also steps through amortization results)
C	Clears the display and the calculation	CLx: clears X only
RM	Recalls M	R↓ (recall M with RCL RM)
orange () orange)	Parentheses	LOG and 10^x (section 5). Roll up with three RM
K	K constant	LAST X
orange SWAP	Swap, or show a second result	$x \geq y$. After a two-result function it shows the second result, which is in Y (section 7.1)
%	$200 \div 15 \text{ %} = \rightarrow 230$	$200 \text{ INPUT } 15 \text{ %} \div \rightarrow 230$
%CHG	$55 \text{ orange } \text{ %CHG } 32 =$	$55 \text{ INPUT } 32 \text{ orange } \text{ %CHG}$
Statistics entry	$x \text{ INPUT } y \text{ } \Sigma+$	$y \text{ INPUT } x \text{ } \Sigma+$ (HP-12C/15C order)
Student's t	df INPUT t blue df,t=P	The same keys; the probability replaces both numbers and the stack drops
TVM, cash flows, bonds, break-even	Store with the key; press the key alone to calculate	The same; a result replaces X (orange IRR/YR and blue ACCRU lift the stack)
Two-result functions ($\bar{x}$, $\bar{y}$, $\hat{y}$, m, SL, NPV, ...)	Main result shown; orange SWAP shows the second	Main result in X, second result in Y; the stack lifts
Numbered registers, STO arithmetic, M	As in the User's Guide	The same (M recalled with RCL RM), plus RCL arithmetic (RCL + n, ...)

The same routines. RPN mode does no arithmetic of its own. It only moves numbers between the stack and the calculator's own routines, so every result matches Chain and Algebraic mode exactly, to all 12 digits. LOG and 10^x , which only RPN mode has, are worked out with the calculator's own logarithm and exponential routines at their full internal precision and rounded once to 12 digits.

10. Getting back to Algebraic or Chain mode, and checking the firmware

Leaving RPN mode. Press **blue** **Alg/Chain** once for **Chain** (**ChAin**) or twice for **Algebraic** (**AlgebrAiC**).

- The number in the display stays.
- The RPN stack is not used in Chain and Algebraic mode. When you come back to RPN, Y, Z, T and LAST X start at zero.
- Registers, M, TVM, cash-flow and statistics data are not affected by switching modes.
- Chain and Algebraic mode work as described in the User's Guide.

Checking the firmware version. Use the built-in self-test:

1. Hold down **ON**, press **PMT**, and release both keys. The display shows **1 ScREEn**.
2. Press **+** twice to reach **3 ChEcKSuM**, then press **=**.
3. Firmware EDED shows **237.237**. (The standard firmware shows **181.181**.)
4. Press **C** to leave the self-test. The calculator stays in the same mode.

HP 10bII+ RPN mode · Prototype firmware EDED.