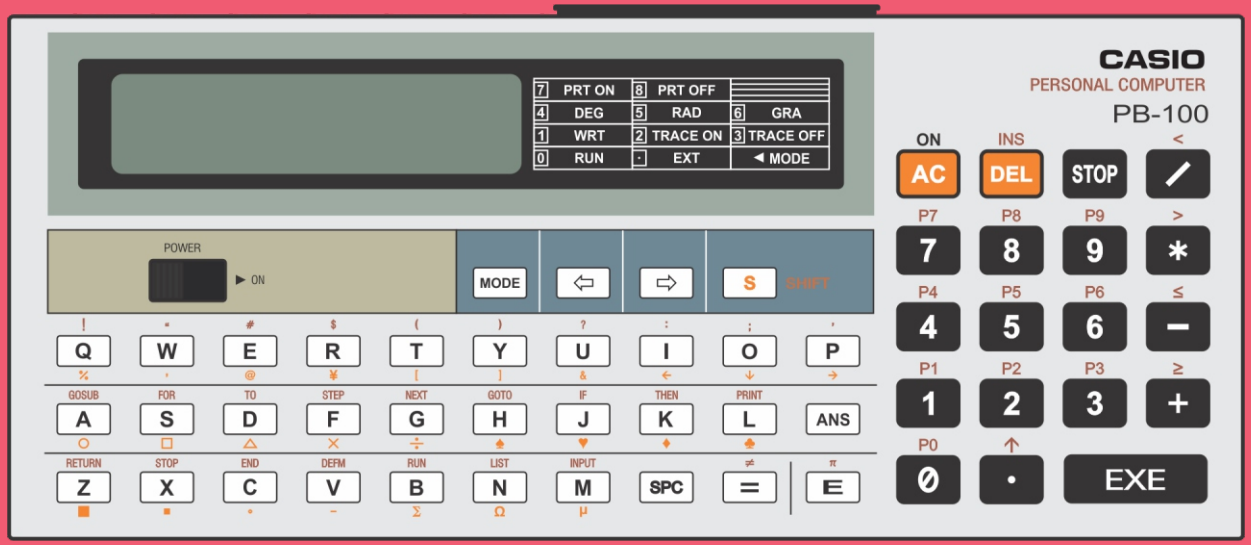
CASIO

PB-100



Command List

* Elements can be used repeatedly
{ } Select one of the elements
[ ] Elements can be omitted

Commands

INPUT

Function: Inputs data from the keyboard to a variable.
Format: INPUT [“message statement”,] variable name *
Examples: INPUT A
INPUT “LOCATION”,C$,”NAME”,D$

PRINT

Function: Displays an output element.
Format: PRINT output element[{, or ;}] *
Examples: PRINT “CASIO”
PRINT A, B, C

GOTO

Function: Branches to a specified location.
Formats: GOTO line number
GOTO #program area number
Examples: GOTO 210
GOTO #4

IF-THEN

Function: When a branching condition is true the statements after THEN are executed.
Formats: IF branching condition THEN statement
IF branching condition THEN branching location
Examples: IF K$=”4" THEN N=N+1
IF A>B THEN 10
IF C=4 THEN #3

FOR-TO-STEP-NEXT

Function: Executes instructions between the FOR and NEXT commands a number of times specified by the control variable.
Format: FOR variable name = numerical expression TO numerical expression [STEP numerical expression] NEXT variable name
Example: FOR X = 0 TO 10 STEP 2
NEXT X

GOSUB

Function: Branches to a specified subroutine.
Formats: GOSUB line number
GOSUB #3
Examples: GOSUB 200
GOSUB #3

RETURN

Function: Returns to the main program from a subroutine.
Format: RETURN
Example: RETURN

STOP

Function: Temporarily stops program execution. Restarts by the ‘EXE’ key.
Format: STOP
Example: STOP

END

Function: Terminates program execution.
Format: END
Example: END

RUN

Function: Executes a program.
Format: RUN [line number]
Examples: RUN
RUN 1000

LIST

Function: Lists the specified program.
Formats: LIST A
LIST [line number]
Examples: LIST A
LIST 100

MODE

Function: Sets the state of the computer.
Formats: MODE 4 (degrees) MODE 5 (radians) MODE 6 (grades)
MODE 7 (PRT) MODE 8 (release PRT)
Example: MODE 4

SET

Function: Specifies the output format for numerical data.
Format: SET {Fn or En or N}
Example: SET F4: PRINT X

VAC

Function: Clears the data in all variables.
Format: VAC
Example: VAC

CLEAR

Function: Erase a program.
Format: CLEAR [A]
Examples: CLEAR
CLEAR A (erases all programs)

DEFM

Function: Expands the number of variables.
Format: DEFM size
Example: DEFM 10

SAVE

Function: Stores a program on cassette.
Format: SAVE [A] [“filename”]
Examples: SAVE “BUDGET”
SAVE A

LOAD

Function: Loads a program from cassette.
Format: LOAD [A] [“filename”]
Examples: LOAD “BUDGET”
LOAD A

PUT

Function: Stores variable data on cassette.
Format: PUT [“filename”] variable1[, variable2] *
Example: PUT “SALES” A, B

GET

Function: Reads variable data from cassette.
Format: GET [“filename”] variable1[, variable2] *
Example: GET “A,B

VER

Function: Verifies the program stored on cassette.
Format: VER [“filename”]
Examples: VER
VER “BUDGET”

Character Functions

KEY

Function: Enters one character from the keyboard.
Format: KEY
Example: A$=KEY

CSR

Function: Displays an output element from a specified position.
Format: CSR numerical expression
Example: PRINT CSRA;”G”;CSR9;”H”

LEN

Function: Provides the length of a character string.
Format: LEN (character string)
Example: X=LEN(B$)

MID

Function: Provides a portion of the special $ character variable.
Format: MID(position[, number of characters])
Examples: A$ = MID(2, 3)
PRINT MID(X,Y)

VAL

Function: Converts a character string into the corresponding numerical value.
Format: VAL (character string)
Example: X=VAL(“123")

Numerical Functions

SIN

Function: Trigonometric sine function (sin X)
Format: SIN (numerical expression)
Example: SIN (A/B)

COS

Function: Trigonometric cosine function (cos X)
Format: COS (numerical expression)
Example: COS (A*10)

TAN

Function: Trigonometric tangent function (tan X)
Format: TAN (numerical expression)
Example: TAN (PI/6)

ASN

Function: Inverse trigonometric sine function (arcsin or sin-1)
Format: ASN (numerical expression)
Example: ASN (X*X)

ACS

Function: Inverse trigonometric cosine function (arccos or cos-1)
Format: ACS (numerical expression)
Example: ACS (A+12)

ATN

Function: Inverse trigonometric tangent function (arctan or tan-1)
Format: ATN (numerical expression)
Example: ATN (A/100)

LOG

Function: Common logarithmic function
Format: LOG (numerical expression)
Example: LOG (2.71828)

LN

Function: Natural logarithmic function
Format: LN (numerical expression)
Example: LN (1.6754)

EXP

Function: Exponential function
Format: EXP (numerical expression)
Example: EXP (1)

SQR

Function: Square root
Format: SQR (numerical expression)
Example: SQR (30)

ABS

Function: Gives the absolute value of the numerical expression.
Format: ABS (numerical expression)
Example: ABS (-10.5)

SGN

Function: Gives the sign of the numerical expression.
Format: SGN (numerical expression)
Example: SGN (-1)

INT

Function: Gives the largest integer which is less than or equal to the specified numerical expression.
Format: INT (numerical expression)
Example: INT (3.14)

FRAC

Function:	Gives the value of the fractional part of the numerical expression.
Format:	FRAC (numerical expression)
Example:	FRAC (2.64)

RND

Function:	Gives the value obtained by rounding the specified digit.
Format:	ROUND (numerical expression, digit position)
Example:	ROUND (1.414, 2)

RAN#

Function:	Gives a random number from 0 to 1.
Format:	RAN#
Example:	INT (RAN# * 10)

Error Message Table

Error	Meaning	Cause
1	Memory overflow or system stack overflow	- Number of steps are insufficient. Program cannot be written. - Stack overflow due to a complicated calculation formula.
2	Syntax error	- Format error in program. - Left-hand and right-hand formats differ in an assignment statement.
3	Mathematical error	- The result of a numerical expression calculation exceeds 10^100 or greater. - The argument of numerical function is outside the input range. - Result is indefinite or impossible.
4	Undefined line number error	- No designated line number for GOTO or GOSUB statement.
5	Argument error	- For a command or function that requires an argument, the argument is outside the input range.
6	Variable error	- Attempt was made to use memory which has not been expanded. - Attempt was made to use the same memory for a numerical variable and a character variable at the same time.
7	Nesting error	- RETURN statement is executed when subroutine is not being executed. - NEXT statement is executed when not in FOR loop. - Subroutine nesting levels exceed 8. - FOR-NEXT loop nesting levels exceed 4.
9	Option error	- Execution is performed in the PRT mode or option command such as SAVE is executed when no printer or cassette recorder is connected.

Specifications

- Calculation Range

+-1 x 10^99 to +_9.999999999 x 10^99 and 0 (internal calculations use 12-digit mantissa)

- Number of steps

Maximum 544 steps (maximum 1,568 steps when optional RAM pack is loaded)

- Program capacity

Maximum 10 programs (P0 through P9)

- Number of variables

Standard 26, expandable to 94 (maximum 222 variables when optional RAM pack is loaded) and exclusive character variable ($)

- Nesting

Subroutine - 8 levels
FOR-NEXT loop - 4 levels
Numerical value - 6 levels
Calculation elements - 12 levels

- Display system and contents

10-digit mantissa (including minus sign) or 8-digit mantissa (7 digits for negative number) and 2-digit exponent.

- Power supply

2 lithium batteries (CR2032)

- Power consumption

Maximum 0.02W

- Battery life (Continuous use)

Mainframe only - approximately 360 hours

- Auto power-off

Power is turned off automatically approximately 7 minutes after last operation.

Character Code Table

	SPACE	+	-	*	/	↑	!	″	#	$	>	≥	=	≤	<	≠
Numbers	0	1	2	3	4	5	6	7	8	9	.	π	)	(	Ē	E
Capital letters	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
	Q	R	S	T	U	V	W	X	Y	Z						
Small letters	a	b	c	d	e	f	g	h	I	j	k	l	m	n	o	p
	q	r	s	t	u	v	w	x	y	z						
Symbols	?	,	;	:												
Graphic symbols	◯	∑	o	Δ	@	X	÷	♠	←	♥	♦	♣	μ	Ω	↓	→
	%	¥	□	[	&	_	‘	·	]	■						

CASIO

Digital Reconstruction: Steven Fox 2021