BASIC Comparison Sheet

Marcus von Cube <marcus@mvcsys.de>

History

06.01.2009 Casio and Sharp Prog/Vars/Mem page
08.01.2009 HP-71, TI-74, CC-40 added
10.01.2009 Some clarifications, functions page started, HP-75
15.01.2009 Functions completed, commands page started
26.01.2009 Output commands, including graphics
31.01.2009 almost complete
02.02.2009 Some more details about files
08.02.2009 Epson HX-20 & TRS-80 Model 100
04.03.2009 Sharp PC-1403
22.03.2009 Canon X-07
12.04.2009 Casio FP-200
15.04.2009 Casio BASIC ROM OM53-B for PB-2000C/AI-1000, USING fixed for PB-1000 and OM-53B
02.08.2009 Minor bugs fixed (Sharp and CURSOR command)
09.10.2009 Differences between Sharp PC-E500 and PC-E500S, extra page for Casio MODE commands
14.12.2009 Some clarifications on PEEK/POKE for low end Sharps
19.01.2011 Sharp PC-1350/PC1421 added.
14.09.2012 Minor corrections.
18.11.2012 Sharp PC-2500
01.04.2012 Corrections to Casio FX-850/880 (AND, OR, XOR, NOT)
29.09.2012 Corrections for display resolution of Sharp PC-E500 and PC-G850

Introduction

This document tries to compare the different BASIC dialects offered by various pocket computers made by Casio, Sharp, HP, TI and other vendors. It is work in progress. Any comments and additions are welcome.

I'm the author of the CASsette I/O Utilities which enable the access to files created by various Casio and some other BASIC pocket computers and transferred via the sound card or a floppy drive to your desktop system. The package includes programs which understand and decipher the tokenized form of a stored BASIC program or create it from a BASIC source text stored on your PC or Mac. I realized that there are many differences between the implementations, ranging from a few minor annoyances to huge gaps in functionality. Here I want to share my findings in table form.

The document will always be incomplete in several ways: I only have access to a limited number of different machines, and I do not plan to cover every aspect of each implementation. I concentrate on areas where the systems are reasonably comparable. This excludes language extensions for special purposes or special software packages. The tables do not try to replace the manuals but will probably aid in finding the correct pages in them.

The Tables

On the following pages you will find the pocket computers in my possession or from which I do have a manual, compared in different areas. There are many more variants of these machines with different memory configurations or some additional features. If you have information or a manual, just send a copy!

Blue entries are manual (non programmable) commands, or mark an example.
Grey entries are either not available or obsolete.
Green arguments are optional and have defaults.
Workarounds replace functions which are not implemented.

1. ProgVarsMem compares program editing, variables and memory organization.
2. Functions compares the built in functions and operators (strings, math, etc.).
3. Commands compares program flow, subroutines, error handling.
4. Special commands collects non obvious information from diverse areas.
5. Casio MODE explains special variants of the MODE command for some Casio models.

Programs, Variables and Memory

Vendor	Casio											
Model	**FX-702**	**PB-100 PB-300 FX-700P FX-710P**	**PB-220 FX-720P**	**FX-730P FX-770P FX-785P FX-790P FX-795P**	**PB-700**	**PB-770**	**FX-750P**	**FX-850P FX-880P**	**VX-4**	**Z-1GR**	**PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B**	**FP-200**
Syntax specifics	Spaces are insignificant. All keywords and variables must be entered in uppercase letters.							Spaces are insignificant except before TO, THEN or ELSE following a variable name. Mixed case entry is allowed and sometimes significant.				
Abbreviations	N/A											Some commands like P. for PRINT
Line numbers	1-9999							1-65535				1-64999
Line length	63				79			255				255
Comments	Comments are N/A	REM, rest of line is ignored.						REM or single quote "'" which implies end of statement (no colon needed), rest ignored.				
Program areas	0-9, switched with keyboard function P0-P9. In MODE 0, program is automatically started.				0-9, PROG n selects area. Keyboard functions P0-P9 start program.			0-9, switched with keyboard function P0-P9. In MODE 0, program is automatically started. Z-1GR uses CAL key instead of MODE 0.			RAM file system with menu selection, BAS files can be run directly.	0-9, PROG n selects area.
Internal file system (more infos on Commands page)	N/A	MEMO database.			N/A			MEMO database.	ASCII file areas F0-F9. F0 is the default MEMO area, also accessible with MEMO key.	RAM FS with Load/Save to area.		10 CETL tables.
Edit programs	MODE 1 sets WRT mode. Select area with keyboard function P0-P9. LIST displays lines for editing.				PROG n selects area for editing. EDIT displays line for editing.			MODE 1 sets BASIC mode. Select area with keyboard function P0-P9. EDIT displays lines for editing.		Menu selects BASIC mode. EDIT displays lines for editing.		EDIT displays line for editing.
								No file system	ASCII file editor			No file system
AUTO, DELETE, RENUM	N/A				AUTO and DELETE			N/A	DELETE and RENUMBER		DELETE RENUM (OM-53B)	RENUMBER
Kill program(s)	CLR, CLR ALL	CLEAR, CLEAR A	NEW kills program in current area, NEW ALL kills all memory.									NEW, NEW ALL clear single area or all areas.
			No file system							Files can be killed from menu.		RESET clears all memory.
Show memory	MODE 1 displays free program memory in special display area, display may fail if free memory greater than 19999 on FX-730P				SYSTEM displays program areas, angle mode and free memory			FRE <n>, n=0..2 0: free variable space, 1: free program space, 2: total variable space.	SYSTEM displays print mode (angle mode on PB-1000), CLEAR parameters and free space. FRE <n>, n=1..5 1, 2: see left, 3: total string space, 4: free variable space, 5: free string space.	FRE <n>, n=1..6 1-5: see left 6: ml space (uncertain).	OM-53B: [memory] menu sets size for file system.	Switching modes between CETL and BASIC shows free memory. SYSTEM shows program sizes for all areas. FRE(dummy number) returns free space for programs. FRE("dummy string") returns free string space.
Memory allocation	DEFM <blocks> allocates additional variables in blocks of 10. Block 1 is A0-A9, Block 20 is T0-T9. 80 bytes are taken from program memory per block.	DEFM <n>, manual mode only. 8 bytes per variable.	DEFM <n> allocates n additional variables. 8 bytes per variable are taken from program memory.		Automatic allocation of two letter "registered" variables			CLEAR <size> allocates space for all variables and strings	CLEAR <str>,<var> allocates <str> bytes for strings, <var> bytes for variables	CLEAR <str>,<ml>,<sys> allocates <str> bytes for strings, <ml> bytes for machine language, <sys> bytes for system (uncertain for Z-1) PB-1000: <sys> is file system size OM-53B: <sys> is not allowed.		AREA <size> sets the size for CETL tables and I/O buffers. CLEAR <str>,<himem> clears all variables, sets size of string area and the high memory limit for BAISC. Above machine language programs can be installed.
Show variable allocation	N/A	DEFM displays variable allocation and switches to DEFM mode			LISTV lists names of arrays and registered variables			VARLIST lists names of arrays and variables				N/A
		No DIM mode	LISTV lists array names in DIM mode									

Programs, Variables and Memory

Vendor	Casio										
Model	**FX-702**	**PB-100 PB-300 FX-700P FX-710P**	**PB-220 FX-720P**	**FX-730P FX-770P FX-785P FX-790P FX-795P**	**PB-700**	**PB-770**	**FX-750P**	**FX-850P FX-880P**	**VX-4**	**Z-1GR**	**PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B**	**FP-200**
Variables	A-Z, A0-A9..T0-T9.	Fixed variables A-Z.			Fixed variables A-Z, maximum of 40 "registered" two letter variables or arrays. Use of undefined two letter variables results in an UV error.			Long variable names, case sensitive, must not start with a keyword. FX-850P/880P restricts length to 15 characters. The limit for VX-4, Z-1, PB-1000, OM-53B is unknown but higher than 64. Longer names are truncated. All variables are independent. Undefined variables return 0.				Long variable names, case sensitive, must not start with a keyword. Maximum length is 255. All variables are independent. Undefined variables return 0. DEFDBL, DEFSGL <letter range> and suffixes #and ! allow double or single precision. Variables of different types are independent from each other.
Numeric range	BCD, 10 digits, 2 digits exponent. Special symbol for the exponent.				BCD, 10 or 5 digits, 2 digits exponent			BCD, 10 digits, 2 digits exponent.				BCD with 2 digts exponent. Single precision: 6+3 digits. Doube precision: 16+3 digits. The three guard digits are not displayed.
String variables	Fixed string variables A$-Z$, 7 characters, share memory with A-Z.							String variable names follow the rules of numeric variables+$. String length is dynamic up to 255 characters. The total amount of string space must be set by CLEAR. Undefined string variables return an empty string.				See left. DEFSTR <letter range> allows names without $.
	$, 30 characters.			$, 62 characters.	Registered two letter (+$) string variables, 16 characters. Use of undefined two letter string variables results in an error.							
String too long	ERR 6			Silent truncation	ST error							
String literals	"STRING"	"String"					"STRING"	"String", "String – Trailing quote can be omitted on line end.				
Character set	FX-702P	PB-100	PB-100 extended		ASCII + japanese + graphics		ASCII	ASCII + japanese + symbols			Same as PB-700	ASCII + symbols
Lower case	NO	Yes					NO	Yes				Yes
Arrays	A(...) overlaps variables A0, A1, etc., created by DEFM <blocks>.	A(...)-Z(...) overlap A-Z: A(0)-A(25) are A-Z, B(0)-B(24) are B-Z, ... Z(0) is Z.		DIM A(...) creates independent array. Maximum of 8 arrays.	DIM A(...) defines normal array, two letter names allowed. DIM A!(...) defines "half-precision" array, only A!-Z! allowed.			DIM Name(...) defines array. Array names follow the rules of numeric variables.				
String arrays	N/A	A$(...)-Z$(...) overlap the fixed string variables A$-Z$ (see above.)		DIM A$(...) creates independent string array, 7 characters.	DIM A$(...)*L defines string array with defined length L, which defaults to 16 characters. Only A$-Z$ allowed. Length can be an expression. Maximum string length is 80.			DIM Name$(...) defines string array. String array names follow the rules of string variables. String length is dynamic up to 255 characters. The total amount of string space must be set by CLEAR.				
Array dimensions	1: (0..10*<blocks>-1) 2: (0..9,0..<blocks>-1)	Maximum index of Z(...) defined by DEFM.		Three dimensions, up to 255. Dimensions can be expressions.	Two dimensions, up to 255. Dimensions can be expressions.			Number and size of dimensions is only limited by memory. Dimensions can be expressions.				Up to three dimensions. Size is limited by memory. OPTION BASE 0 or 1 set the lowest index for next DIM.
Automatic DIM	No DIM mode.			DEFM works like PB-100. DIM disables DEFM mode.	All arrays must be defined.			DEFM works like PB-100. DIM disables DEFM mode.	All arrays must be defined.			Arrays of dimension (10), (10,10) or (10,10,10) can be created implicitly by an assignment.
Clear variables	VAC resets all variables.	CLEAR or VAC resets all fixed variables.			CLEAR resets all fixed variables and deletes all registered variables and arrays.			CLEAR or VAC deletes all variables and arrays.	CLEAR deletes all variables and arrays.			
		No DIM mode.		DIM mode: CLEAR deletes all arrays, ERASE deletes selected arrays.	ERASE deletes selected registered variables or arrays.			ERASE deletes selected arrays.				N/A

Programs, Variables and Memory

Vendor	Sharp											
Model	**PC-1500A**	**PC-1210 PC-1211 PC-1212**	**PC-1245 PC-1246 PC-1247 PC-1248 PC-1251**	**PC-1401 PC-1421**	**PC-1403**	**PC-1260 PC-1261 PC-1262**	**PC-1350 PC-1360 PC-2500**	**PC-1280**	**PC-1475**	**PC-E220 PC-G820**	**PC-G850**	**PC-E500 PC-E500S**
Syntax specifics	Spaces are insignificant. All keywords and variables must be entered in uppercase letters.							Spaces are insignificant. Mixed case entry is allowed but insignificant.				
Abbreviations	Keywords can be abbreviated with "."											
Line numbers	1-65279	1-999	1-65279									
Line length	79 (keywords single char)	79 + line number (keywords count as single char)								255		
Comments	REM, rest of line is ignored.									REM or single quote "'" which implies end of statement, rest ignored.		
Program areas	Single program, labels allow direct access with DEF key.									RAM file system. FILES shows filenames. Shift+LOAD loads file (L on PC-E500)		
Internal file system (more infos on Commands page)	No file system						RAM cards			No drive name		"E:" internal RAM, "F:" RAM card.
							No file system	Card file system "F:"				
Edit programs	PRO(GRAM) mode must be selected by switch or BASIC or MODE key, LIST displays lines for editing.											
	No file system						TEXT switches to text mode, BASIC switches back.			ASCII file editor		TEXT switches to text mode, BASIC swiches back.
AUTO, DELETE, RENUM	N/A				DELETE, RENUM	N/A	PC-1360: DELETE, RENUM	AUTO, DELETE, RENUM		DELETE, RENUM		AUTO, DELETE, RENUM
Kill program(s)	NEW kills unprotected program or RSV area. NEW0 clears everything, even a protected program.			NEW deletes currently loaded program.								
				N/A		NEW# kills ESP.	N/A	KILL"F:file.BAS"		KILL"file.BAS", Kill from editor menu.		KILL"E:..." KILL"F:..."
Show memory	MEM returns free memory.	MEM is command.	MEM returns free memory.							FRE returns free memory, including free RAM disk space		FRE<n> n=0,1 returns free space. n=1 reorganizes string space.
	STATUS <n>, n=1..4 1: free memory 2: used memory 3: end of of program 4: last line executed	N/A				MEM# returns free ESP memory.	N/A	DSKF3 returns free space on RAM card file system				DSKF<d> returns free disk space, d=3, 4 or "E:", "F:"
Memory allocation	Automatic allocation of two letter variables.	Assigning a value to A(n) or A$(n) with n>26 reserves space. 8 bytes per variable are taken from program memory.								MON starts machine language monitor, which can set aside memory with USER command. *USER <end> sets end address to value <end> above 00FF. A value of 00FF cancels reservation		MEM$="X", "S1": internal RAM, RAM card is disk "F" "S2": RAM card only, no disk "F:" "B": all RAM is merged, no disk "F:"
		N/A				EQU# n reserves n additional 128 byte blocks for ESP usage. BASIC area must be empty.	SET MEM "X" (MEM "X" on PC-1350) "B": all memory is merged, no RAM disk. PC-1350 only: "C": use card for program and variables PC-1360 only: "C": only one card inserted. "D": program in card 1, data in card 2. PC-1280 / PC-1475 only: "1": PC-1475 with one card or PC-1280 with internal memory only, RAM disk "F:" on spare card "2": internal ram or card 1 unused CONVERT n (PC-1475) migrates cards.				
Show variable allocation	N/A					EQU# returns n.	MEM$ returns configuration			*USER in ML monitor		MEM$ returns configuration

Programs, Variables and Memory

<table>
<tr><th>Vendor</th><th colspan="11">Sharp</th></tr>
<tr><th>Model</th><th>PC-1500A</th><th>PC-1210
PC-1211
PC-1212</th><th>PC-1245
PC-1246
PC-1247
PC-1248
PC-1251</th><th>PC-1401
PC-1421</th><th>PC-1403</th><th>PC-1260
PC-1261
PC-1262</th><th>PC-1350
PC-1360
PC-2500</th><th>PC-1280</th><th>PC-1475</th><th>PC-E220
PC-G820</th><th>PC-G850</th><th>PC-E500
PC-E500S</th></tr>
<tr><td>Variables</td><td rowspan="2">Fixed variables A-Z, automatic two letter variables. Longer names are truncated. Undefined variables return 0.</td><td rowspan="2">Fixed variables A-Z.</td><td colspan="6">Fixed variables A-Z, automatic independent two letter variables. Longer names are truncated. Undefined variables return 0.
PC-1421 has special variables for financial calculations:
n, i, PV, FV, PMT, CST, SEL, MAR, MU, NPV, IRR, PRN, INTE, BAL, SPRN, SINTE</td><td colspan="4">Long variable names, case insensitive, up to 32 chars. Must not start with keyword. Longer names are truncated. All variables are automatic and independent. Undefined variables return 0.</td></tr>
<tr><td colspan="5">Single precision only</td><td>DEFDBL, DEFSGL and suffixes # and ! allow double and single precision.
DP variables are always independent. A!-Z! are identical to fixed variables A-Z.</td><td rowspan="2">Single precision only</td><td colspan="2">DEFDBL, DEFSGL and suffixes # and ! allow double and single precision. Single and double precision variables are independent from each other.</td></tr>
<tr><td colspan="5">DEFDBL/DEFSGL set default.</td><td colspan="2">Default is set with DEFDBL/DEFSGL.</td></tr>
<tr><td>Numeric range</td><td colspan="6">BCD, 10 digits, 2 digits exponent. Internal accuracy up to 12 digits.
The PC-1211/1248 have a special symbol for the exponent (leading 1 can be omitted)</td><td>BCD,
10 digits or 20 digits,
2 digits exponent.</td><td>BCD, 10 digits,
2 digits exponent.</td><td colspan="2">BCD, 10 digits or 20 digits, 2 digits exponent.</td></tr>
<tr><td>String variables</td><td rowspan="2">Fixed string variables A$-Z$, independent of A-Z.
Automatic two letter (+$) string variables, 16 characters for all types.
Undefined string variables return an empty string.</td><td colspan="10">Fixed string variables A$-Z$, 7 characters, share memory with A-Z.</td></tr>
<tr><td>N/A</td><td colspan="9">Automatic two letter (+$) string variables, 16 characters. Undefined string variables return an empty string.</td><td>String variable names follow the rules of numeric variables+$.
String length is dynamic up to 254 characters. Undefined string variables return an empty string.</td></tr>
<tr><td>String too long</td><td colspan="11">Silent truncation</td></tr>
<tr><td>String literals</td><td>"String", "String</td><td colspan="3">"STRING", "STRING</td><td colspan="7">"String", "String – Trailing quote can be omitted on line end on all Sharp BASIC implementations</td></tr>
<tr><td>Character set</td><td colspan="3">ASCII + symbols</td><td colspan="7">ASCII + japanese + symbols</td><td>IBM-PC code page 437</td></tr>
<tr><td>Lower case</td><td>Yes</td><td colspan="3">NO</td><td colspan="7">Yes</td></tr>
<tr><td>Arrays</td><td rowspan="2">DIM Name(...) defines array. Name is one or two characters.</td><td rowspan="2">Only A() allowed to access A-Z.</td><td rowspan="2">DIM B(...)-Z() define arrays. A() is reserved.</td><td colspan="6">DIM Name(...) defines array. Name is one or two characters.
PC-1421 has special arrays CFI and NFI with index 0..19.</td><td>DIM Name(...) defines array.
Array names follow the rules of numeric variables.</td></tr>
<tr><td colspan="4">Single precision only.</td><td colspan="2">Single or double precision arrays are independent</td><td>Single precision only.</td><td>Single and double precision arrays are independent from each other.</td></tr>
<tr><td>String arrays</td><td>DIM Name$(...)*L creates string array. See right for details.</td><td>Only A$() allowed to access A$-Z$.</td><td colspan="8">DIM Name$(...)*L creates string array with maximum string length L which defaults to 16.
Names follow the rules of numeric arrays+$.
Length can be an expression. Maximum string length is 80.</td><td>DIM Name$(...) defines string array.
String array names follow the rules of string variables.
String length is dynamic up to 254 characters.</td></tr>
<tr><td>Array dimensions</td><td>Two dimensions up to 255. Dimensions can be expressions.</td><td>N/A</td><td colspan="8">Two dimensions up to 255. Dimensions can be expressions.</td><td>Number and size of dimensions is only limited by memory. Dimensions can be expressions.</td></tr>
<tr><td>Automatic DIM</td><td>@(1) to @(26) address variables A-Z.</td><td colspan="9">A(1) to A(26) or A$(1) to A$(26) can be used to address variables A-Z if they contain the proper data type. Assigning a value to higher index values reserves additional space.
DIM A(...) or A$(...) disables this overlap except on PC-121x and PC-1248 which reserve A(...).</td><td>All arrays must be defined.</td></tr>
<tr><td rowspan="2">Clear variables</td><td colspan="10">CLEAR resets all fixed variables and deletes all automatic variables and arrays.</td><td>CLEAR deletes all variables and arrays.</td></tr>
<tr><td colspan="5">PC-1421: ERASE FIN clears financial variables, ERASE deletes selected arrays.
All others: You cannot erase a single array or variable.</td><td colspan="7">ERASE deletes selected arrays (except PC-1350.)</td></tr>
</table>

Programs, Variables and Memory

Vendor	HP		TI		Tandy Radio Shack	Canon	Epson
Model	**HP-75**	**HP-71**	**TI-74**	**CC-40**	**TRS-80 Mod. 100**	**X-07**	**HX-20**
Syntax specifics	Spaces are insignificant. Mixed case entry is allowed but insignificant. Statement delimiter is @ instead of :		Spaces or other delimiters needed around keywords. Mixed case entry is allowed but insignificant.		Spaces are irrelevant but kept by the tokenizer. Mixed case is allowed but insignificant.		Spaces are relevant before some keywords like TO. Spaces are kept in the code by the tokenizer. Mixed case is allowed but insinificant.
Abbreviations	With "."	N/A	Some keywords have short forms.		? is an abbreviation for PRINT.		
Line numbers	1-9999		1-32766		1-65529		1-63999
Line length	95		80		254	80 (editor restriction)	255
Comments	REM or "**!**" which implies end of statement (no colon needed), rest of line ignored.				REM or single quote "'" which implies end of statement, rest ignored.		
Program areas	RAM file system + one workfile. EDIT <file> selects current file. RUN <file> sets current file and starts it.		Only a single program.		RAM file system (6+2) with menu. LOAD or SAVE set a pointer to the current file instead of making a copy. Editing the current program changes the file in the RAM filesystem. Tokenized BASIC programs can be run from the menu. A single unnamed BASIC program can exist besides the named files.	RAM file system (6+1). RAM cards occupy the top of RAM and can be used as removable file storage. RUN "file" starts program without loading it into working storage.	5 login areas, switched with LOGIN n, n=1..5. Programs can be given a TITLE and appear in the start menu.
Internal file system (more infos on Commands page)	Internal RAM file system. Ports N/A. No info about modules available.	Module ports 0 to 5. Large modules are divided in sub ports: 5.03. Syntax is "FILE:PORT(n)" or "FILE:MAIN". Quotes around filenames are optional.	RAM module. Memory can be swapped or copied with CALL GET(...) and CALL PUT(...).	Memory swapping with PUT and GET is N/A.			
Edit programs	EDIT 'file', BASIC|TEXT FETCH	EDIT <file> creates or selects file. Up and down arrow select lines for editing. FETCH <line>/<label> selects specific line.	LIST <line> or up and down arrow display lines for editing. Lines must be deleted using DEL, simple entry of a number is treated as a computation.		EDIT first-last starts full screen editor. "." is the last line edited. The command creates a temporary ASCII file that is merged upon editor exit.	LIST first-last lists line on screen for full screen editor. Screen size is just 80 characters. Use LIST@ line + ON/BREAK key to edit lines longer than 60 characters.	LIST first-last displays lines which can be edited on the full virtual screen. "." is the last line edited.
AUTO, DELETE, RENUM	AUTO, DELETE, RENUMBER		NUM/NUMBER, DEL/DELETE, REN/RENUMBER		DELETE	N/A	AUTO/DELETE/RENUM
Kill program(s)	DELETE ALL deletes all lines in current file.		NEW deletes program and variables. NEW ALL clears all memory (total reset). CALL ADDMEM (see below) forces NEW ALL.		NEW deletes current program and variables.		
	PURGE <file> deletes file from internal RAM, port or device.				KILL"file" kills file from RAM disk.	DELETE "file","type" kills file from RAM disk.	Shift+Ctrl+3 in startup menu forces memory clear (after ENTER).
Show memory	MEM returns free memory.		FRE(n), n=0,1 0: total user memory, 1: space used by program and variables.	FRE(n), n=0..5 0, 1: see left 2: free + temporary mem 3: largest block size 4: free memory 5: # of free blocks	FRE(dummy number) returns free space for programs. FRE("dummy string") returns free string space.		
	Ports N/A.	MEM(port) returns free space in specified port. SHOWPORT lists available ports.			MAXRAM returns the highest available memory address. HIMEM returns the currently set upper memory address for BASIC or files.	The manual documents all system pointers. Use PEEK to get the corresponding values.	STAT area prints size of current or selected program. STAT ALL prints a complete overview.
Memory allocation		CLAIM PORT(port) and FREE PORT(port) add or remove port memory to main memory.	CALL ADDMEM adds RAM module to user memory. NEW ALL releases the RAM module. Both commands clear all memory.		CLEAR <str>,<himem> clears all variables, sets size of string area and the HIMEM value. Use MAXRAM as the second argument to recover all available RAM. The RAM file system works "in place", so editing any file, BASIC or TEXT, moves memory around and affects the free space. Machine language programs are copied to their saved memory location and can only be loaded if enough high memory is reserved.	CLEAR <str>,<himem> clears all variables an sets size of the string area and the BASIC upper memory limit which is below the file area. FSET <size> reserves memory for the file area at top of memory. If the size is less or equal to the size of an inserted RAM card, this card can be used as a removal filesystem. If the power up routine detects a configuration change (e. g. RAM card swap) you are prompted to allow adjustment of the system pointers. So cards of different size can be used alternately.	CLEAR <str>,<RAM file> clears all variables, sets size of string area and size of the RAM file. MEMSET <address> sets aside low memory for machine language programs. Default address is &H0A40. WIDTH <cols>,<rows>,<margin> allocates the virtual screen area and affects the free space.
			No machine language support.	CALL GETMEM(size,ptr) returns a free memory block for machine language use. Variable ptr is initialized with the base address. CALL RELMEM(adr) releases the block.			
Show variable allocation		SHOW PORT list available ports.	See FRE(...)		See FRE(...)		

Programs, Variables and Memory

Vendor	HP		TI		Tandy Radio Shack	Canon	Epson
Model	**HP-75**	**HP-71**	**TI-74**	**CC-40**	**TRS-80 Mod. 100**	**X-07**	**HX-20**
Variables	A-Z, A0-A9..Z0-Z9. Variables are independent but arrays and simple variables cannot share the same name. Undefined variables return 0. Variables are local to the running program or procedure.		Long variable names, case insensitive, up to 15 chars, @ and _ allowed. Variables may even start with a keyword. Longer names result in error. All variables are automatic and independent. Variables are created by a manual assignment or automatically by RUN if used in a program. Variables are created before the program is executed, so all used variables are defined. Use of undefined variables in command mode results in an error. Variables in procedures are local to the procedure but can be made persistent between calls by ATTACH <procedure>.		Two character variable names, case insensitive. Must not contain keyword. Longer names are truncated. All variables are automatic and independent. Undefined variables return 0.		Long variable names, case insensitive, up to 32 chars. Must not start with keyword. Longer names are truncated. All variables are automatic and independent. Undefined variables return 0.
	REAL, SHORT and INTEGER declare variables with standard, half or integer precision.				DEFDBL, DEFSGL, DEFINT <letter range> and suffixes #, ! and % allow double or single precision and 16 bit signed integers. Variables of different types are independent from each other.		
	Default is REAL.				Default is double precision.	Default is double precision. All float constants are double precision.	Default is single precision.
Numeric range	Like HP-71 but not IEEE. SHORT exponent range +/-99.	BCD, 12 digits or 5 digits, exponent range +/-499. Special values for *NaN* and *Inf* (IEEE standard). Integers are BCD with 5 digits and no exponent. Some internal computations use 15 digits.	BCD, 13 digits, exponent range -128..+127.		Integer: 16 bit signed, -32768..32767. Single precision: 32 bit binary float (ca.7 digits), exponent range -64..+62. Double precision: 64 bit binary float (ca.14 digits), exponent range -64..+62. Double precision numbers have a D exponent or a trailing #.		
String variables	A$-Z$, A0$-A9$..Z0$-Z9$. DIM A$[L] sets string length to L. Default is 32, maximum is memory dependent. String allocation is static. String variables are independent but arrays and simple variables cannot share the same name. Undefined string variables return an empty string.		String variable names follow the rules of numeric variables+$. String length is dynamic up to 255 characters. String variables are created by a manual assignment or automatically by RUN if used in a program. Use of undefined string variables results in an error.		String variable names follow the rules of numeric variables+$. String length is dynamic up to 255 characters. Undefined string variables return an empty string. DEFSTR <letter range> allows names without $.		
String too long	ERR:Excess Chars		E3 Mismatch		LS Error		
String literals	"String with 'quotes' ", 'String with "quotes" '		"String with ""quotes"" "		"String", quotes inside string literals are not supported.		
Character set	ASCII + symbols				ASCII + symbols.		National ASCII + symbols.
Lower case	Yes				Yes		
Arrays	DIM A(...) defines array. REAL, SHORT or INTEGER A(...) define arrays of certain type. See above for naming restrictions.		DIM Name(...) defines array. Array names follow the rules of numeric variables. DIM statements are static declarations and must appear above any reference to the array in the program. DIM cannot appear after THEN or ELSE. After a DIM statement only comments are allowed on the same line.		DIM Name(...) defines array. Array names follow the rules of numeric variables.		
	Static declaration like TI-74.	Dimension and size can be expressions. Existing arrays can be redimensioned without data loss.			Arrays of different types are independent from each other.		
String arrays	N/A	DIM A$(n)[L] defines string array and sets string length to L. See above for naming restrictions. Only one dimension allowed. Default length is 32, maximum is memory dependent.	DIM Name$(...) defines string array. Array names follow the rules of numeric variables. String length is dynamic up to 255 characters. See above for DIM statement restrictions.		DIM Name$(...) defines string array. String array names follow the rules of string variables. String length is dynamic up to 255 characters.		
Array dimensions	Two dimensions. Size is only limited by memory. OPTION BASE 0 or 1 set the lowest index for next DIM statement.		Three dimensions. Size is only limited by memory. Dimensions must be constant.		Number and size of dimensions is only limited by memory. Dimensions can be expressions.		Limits see left. OPTION BASE 0 or 1 set the lowest index for all arrays.
Automatic DIM	Arrays of dimension (10) or (10,10) can be created implicitly by an assignment.		Arrays of dimension (10), (10,10) or (10,10,10) can be created implicitly by an assignment.		Arrays of dimension (10), (10,10) or (10,10,10) or more can be created implicitly by an assignment.		
Clear variables	CLEAR VARS	DESTROY ALL deletes all variables and arrays.	Program editing, power cycling, NEW or RUN delete all variables.		Power cycling, NEW, RUN or CLEAR delete all variables.		
	N/A	DESTROY deletes selected variables or arrays.	N/A	CALL CLEANUP clears variables not used by the program.	N/A	SLEEP turns power of without deleting the variables. ERASE deletes selected arrays.	ERASE deletes selected arrays.

Functions

	Vendor	Casio										
	Model	**FX-702**	**PB-100 PB-300 FX-700P FX-710P**	**PB-220 FX-720P**	**FX-730P FX-770P FX-785P FX-790P FX-795P**	**PB-700**	**PB-770**	**FX-750P**	**FX-850P FX-880P**	**VX-4**	**Z-1GR**	**PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B**	**FP-200**
	Syntax remarks	Parentheses around arguments of many functions are optional.											All function arguments need parentheses.
	Precision of mathematical functions	Default precision											Same as argument.
Strings	**Concatenation**	string1 + string2											
	Substrings	MID(start,length)		MID is obsolete		LEFT$(string,length) RIGHT$(string$,length) MID$(string,start,length), length defaults to rest of string.							
		N/A		MID$(start,length)									
		Works only with the special string variable $. Length defaults to rest of string.											
	Length	LEN(string variable)				LEN(string)							
	ASCII to string	N/A, no ASCII code				CHR$(code)							
	String to ASCII					ASC(char)							
	Number to string	N/A		STR$(expression)									
	String to number	N/A	VAL(string variable)			VAL(string), evaluation stops at first illegal character.							
	Expression evaluation	N/A			795P: $="expr" MODE 20,V	N/A			VALF(string)			N/A	
	Search substring	N/A											
	Case conversion												
	Repeat string												
Numbers	**Absolute Value**	ABS x											ABS(x)
	Sign	SGN x											SGN(x)
	Integer part	SGN x * INT ABS x											FIX(x)
	Fractional part	FRAC x											FRAC(x)
	Largest integer below or equal	INT x											INT(x)
	Smallest integer above or equal	-INT -x											-INT(-x)
	Round to d decimal places. Examples round to cents. Workarounds for positive x only!	RND(x,-d – 1), RND(X,-3)				ROUND(x,-d – 1), ROUND(X,-3)							
						N/A			RND is an obsolete alias.			N/A	
		Display only: SET Fd, SET Ed set display precision.				Display only: PRINT USING "###.##";X Works for current line only.			Display only: SET Fd, SET Ed set display precision			Display only: PRINT USING "###.##";X Works for current line only.	
	PI	Symbol π				PI							4*ATN(1) in ANGLE 1
													4*ATN(1#) in ANGLE 1
	Other constants	N/A											
	Maximum	N/A											
	Minimum												
Arithmetic	**Implied multiplication AB=A*B**	No											
	Power x^y	x ↑ y (special symbol)				x ^ y							
	Integer division	INT(a / b)											
	Modulo	a – b * INT(a / b)											
	Remainder	SGN a * (ABS a – ABS b * INT ABS(a / b))				a MOD b							
	Reduction	N/A											
	Percentage	a * p / 100											
Logic	**Comparisons**	<, ≤, >, ≥, =, ≠ (special symbols)				<, <=, =<, >, >=, =>, =, <>, ><							
	Result of 1=1	Comparisons cannot appear outside IF statement							-1, can only be used inside a program, not in direct mode.				-1
	Logical operators	N/A							NOT, AND, OR, XOR				NOT, AND, OR, XOR, EQV, IMP
	Number of bits								16, signed				
	Priority of NOT								Low				Low. Manual recommends parentheses.
Conversions	**HEX format for integers**	N/A			&H0-&HFFFF								N/A
	HEX display				HEX$(n), n<$2^{16}$, result is padded with "0" to four hex digits.								
	Deg/min/sec to decimal	DEG(d,m,s)	N/A	DEG(d,m,s)									
	Number to deg/min/sec	DMS (display)		DMS$(x)									
	Degrees to radians	d / 180 * π				d / 180 * PI							d / 180 * PI (define PI)
	Radians to degrees	r / π * 180				r / PI * 180							r / PI * 180 (define PI)
	Polar to rectangular	PRC r,θ -> x=X,y=Y	x=r * COS θ, y=r * SIN θ		REC(r,θ) -> X,Y	Workaround see left			REC(r,θ) -> x=X, y=Y			Workaround see left	
	Rectangular to polar	RPC x,y -> r=X,θ=Y Commands	r=SQR(x*x + y*y) θ=ACS(x / r)		POL(x,y) -> X,Y See FX-850P.				POL(x,y) -> r=X, θ=Y Both functions return variable X.				
	More conversions	N/A											CDBL, CSNG convert between double and single precision.

Functions

Vendor		Casio											
	Model	FX-702	PB-100 PB-300 FX-700P FX-710P	PB-220 FX-720P	FX-730P FX-770P FX-785P FX-790P FX-795P	PB-700	PB-770	FX-750P	FX-850P FX-880P	VX-4	Z-1GR	PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B	FP-200
Logs, powers, roots	Common logarithm $\log_{10}x$	LOG x				LGT x			LOG x			LGT x	LGT(x)
	Natural logarithm ln x, $\log_e x$	LN x				LOG x			LN x			LOG x	LOG(x)
	More accurate ln(x+1)	LN(x + 1)				LOG(x + 1)			LN(x + 1)			LOG(x+1)	
	Common antilogarithm 10^x	10 ↑ x				10 ^ x							
	Natural antilogarithm e^x	EXP x											EXP(x)
	More accurate e^{x-1}	EXP x – 1											EXP(x) – 1
	Exponent part of number	INT LOG ABS x for x <> 0				INT LGT ABS x for x <> 0			INT LOG ABS x for x <> 0			see PB-700	INT(LGT(ABS(x)))
	Square root √x	SQR x											SQR(x)
	Cube root $\sqrt[3]{x}$	x ↑ (1/3), x ≥ 0			CUR x	x ^ (1/3), x ≥ 0			CUR x			x ^ (1/3)	
	General root $\sqrt[y]{x}$	x ^ (1 / y)											
	Square x^2	x * x											
	Cube x^3	x ↑ 3			CUB x	x ^ 3			CUB x			x ^ 3	
Trigonometrics, hyperbolics	sin x, cos x, tan x	SIN x, COS x, TAN x											SIN(x), COS(x), TAN(x)
	$\sin^{-1} x$	ASN x											ASN(x)
	$\cos^{-1} x$	ACS x											ACS(x)
	$\tan^{-1} x$	ATN x											ATN(x)
	sec x, cosec x, cotan x	1 / COS x, 1 / SIN x, 1 / TAN x											1/COS(x), 1/SIN(x), 1/TAN(x)
	Angle to x-axis	ATN(y / x), result probably in wrong quadrant											
	sinh x	HSN x	(EXP x – EXP(-x)) / 2		HYPSIN x	(EXP x – EXP(-x)) / 2			HYPSIN x				(EXP(x) – EXP(-x)) / 2
	cosh x	HCS x	(EXP x + EXP(-x)) / 2		HYPCOS x	(EXP x + EXP(-x)) / 2			HYPCOS x				(EXP(x) + EXP(-x)) / 2
	cosh x	HTN x	1–2*EXP x/(EXP x+EXP(-x))		HYPTAN x	1–2*EXP x/(EXP x+EXP(-x))			HYPTAN x				1–2*EXP(x)/(EXP(x)+EXP(-x))
	$\sinh^{-1} x$	AHS x	LN(x + SQR(x*x + 1))		HYPASN x	LOG(x + SQR(x*x + 1))			HYPASN x				LN(x + SQR(x*x + 1))
	$\cosh^{-1} x$	AHC x	LN(x + SQR(x*x – 1))		HAPACS x	LOG(x + SQR(x*x – 1))			HYPACS x				LN(x + SQR(x*x – 1))
	$\tanh^{-1} x$	AHT x	LN((1 + x) / (1 – x)) / 2		HYPATN x	LOG((1 + x) / (1 – x)) / 2			HYPATN x				LN((1 + x) / (1 – x)) / 2
	Angle mode degree	MODE 4				ANGLE 0			ANGLE 0, MODE 4			ANGLE 0	
	Angle mode radian	MODE 5				ANGLE 1			ANGLE 1, MODE 5			ANGLE 1	
	Angle mode grad	MODE 6				ANGLE 2			ANGLE 2, MODE 6			ANGLE 2	
Probability	Factorial n!	n! (postfix)	N/A		FACT n	N/A			FACT n			N/A	
	Permutations nPr	n! / (n-r)!			NPR(n,r)				NPR(n,r)				
	Combinations nCr	n! / ((n-r)! * r!)			NCR(n,r)				NCR(n,r)				
	Random number	RAN#				RND			RAN#			RND ctrl ctrl > 0: next in series ctrl = 0: repeat last # ctrl < 0: new series	
	Set random seed	N/A										RND -1 starts new series.	RANDOMIZE RND(-1)
Statistics	Clear statistics registers	SAC	N/A		STAT CLEAR	N/A	STAT CLEAR	Library applications 6500, 6510		STAT CLEAR			
	Add data point	STAT x,y;frq			STAT x,y;frq		STAT x,y;frequency			STAT x,y;frequency			STAT x,y
	Remove data point	DEL x,y;frq			manual only		N/A			N/A			
	List sums and results	N/A			STAT LIST		STAT [L]LIST (sums only)			N/A			
	Sums	CNT, SX, SY, SX2, SY2, SXY			STAT LIST only		CNT, SUMX, SUMY, SUMX2, SUMY2, SUMXY			CNT, SUMX, SUMY, SUMX2, SUMY2, SUMXY			
	Means and standard deviations	MX, SDX, SDXN, MY, SDY, SDYN					MEANX, SDX, SDXN, MEANY, SDY, SDYN			MEANX, SDX, SDXN, MEANY, SDY, SDYN			
	Linear regression coefficients	LRA, LRB, COR					LRA, LRB, COR			LRA, LRB, COR			LRA, LRB
	Linear estimations	EOX y, EOY x			EOX y, EOY x		EOX y, EOY x			EOX y, EOY x			y=LRA*x+LRB x=(y-LRB)/LRA

Functions

Vendor		Casio										
	Model	**FX-702**	**PB-100 PB-300 FX-700P FX-710P**	**PB-220 FX-720P**	**FX-730P FX-770P FX-785P FX-790P FX-795P**	**PB-700**	**PB-770**	**FX-750P**	**FX-850P FX-880P**	**VX-4**	**Z-1GR**	**PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B**	**FP-200**
Various functions	**Set time and date**	N/A										PB-1000 only: TIME$="hh:mm" DATE$="MM-DD-YYYY"	TIME$="hh:mm:ss" DATE$="YY/MM/DD"
	Get time and date											PB-1000: TIME$, DATE$	TIME$, DATE$
	Get timer value	N/A									TIMER	N/A	
	Read memory	N/A			MODE18(a,b$)	N/A	b=PEEK a	N/A	DEFSEG=segment : b=PEEK address / Z-1: INP port				b=PEEK(address)
	Modify memory				MODE19(a,b)		POKE a,b		DEFSEG=segment : POKE address,b / Z-1: OUT port,b				POKE address,b
	Get variable address	Fixed variables are at fixed addresses, only useful, if PEEK/POKE are available.							N/A				
	Call machine language	N/A								MODE110	Z-1/PB-1000: CALL address OM-53B: SYSTEM CALL address PB-1000 only: CALL "ml-file"		CALL address,A,HL,DE,BC
													N/A
	User defined function	N/A											DEF FN X(...)=..., DEF FN X$(...)=... Names follow variable syntax.
	Multi line												
	Recursion												N/A
	Swap Variables	N/A									SWAP	N/A	
	More functions	N/A											CETL access: RC(r),IT(c),FL(f,r,i) SUMRC(r1,r2), SUMIT(i1,i2)

Functions

	Vendor	Sharp										
	Model	**PC-1500A**	**PC-1210 PC-1211 PC-1212**	**PC-1245 PC-1246 PC-1247 PC-1248 PC-1251**	**PC-1401 PC-1421**	**PC-1403**	**PC-1260 PC-1261 PC-1262**	**PC-1350 PC-1360 PC-2500**	**PC-1280**	**PC-1475**	**PC-E220 PC-G820**	**PC-G850**	**PC-E500 PC-E500S**
	Syntax remarks	Parentheses around arguments of many functions are optional. PC-121x allows missing ')'											
	Precision of mathematical functions	Default precision							Selectable		Single precision		Selectable
Strings	**Concatenation**	string1 + string2	N/A	string1 + string2									
	Substrings	LEFT$(s,l) RIGHT$(s,l) MID$(s,st,l)		LEFT$(string,length) RIGHT$(string$,length) MID$(string,start,length)									
	Length	LEN string		LEN string									
	ASCII to string	CHR$ code		CHR$ code									
	String to ASCII	ASC char		ASC char									
	Number to string	STR$ expr		STR$ expression									
	String to number	VAL string, stops at first illegal character.		VAL string, evaluation stops at first illegal character.									
	Expression evaluation			N/A									EVAL string AER n(args)
	Search substring	N/A		N/A									
	Case conversion												
	Repeat string												
Numbers	**Absolute Value**	ABS x											
	Sign	SGN x											
	Integer part	SGN x * INT ABS x											
	Fractional part	SGN x * (ABS x - INT ABS x)											
	Largest integer below or equal	INT x											
	Smallest integer above or equal	-INT -x											
	Round to d decimal places. Examples round to cents. Workarounds for positive x only!	INT(x * 10^d + 0.5) / 10^d, INT(X*100+0.5)/100 PC-1421: see PC-1403.				USING "##.##" MDF X	Use INT (see left)		USING "###.##" sets precision for MDF(X,threshold), threshold defaults to 4.				
		Display only: USING "###.##" sets display format, PRINT USING "###.##";X sets format and outputs X. USING is persistent, even if used together with PRINT.											PRINT USING works for current line only.
	PI	Symbol π											PI, PI#
		PI	N/A	PI					PI, PI#, π#		PI		
	Other constants	N/A											
	Maximum	(a > b) * a + (a <= b) * b									-(a > b) * a - (a <= b) * b		
	Minimum	(a > b) * b + (a <= b) * a									-(a > b) * b - (a <= b) * a		
Arithmetic	**Implied multiplication AB=A*B**	No	Yes, high priority	No									
	Power x^y	x ^ y											
	Integer division	INT(a / b)											
	Modulo	a – b * INT(a / b)											
	Remainder	SGN a * (ABS a – ABS b * INT ABS(a / b))											
	Reduction	N/A											
	Percentage	a * p / 100, short form for PC-1211/1248: AP/E2											
Logic	**Comparisons**	<, <=, >, >=, =, <> PC-121x can compare strings with = only.											
	Result of 1=1	1									-1		
	Logical operators	NOT, AND, OR	1-, *, +	NOT, AND, OR					NOT, AND, OR, XOR (PC-E220 lacks XOR)				
	Number of bits	16, signed	logical only	16, signed									
	Priority of NOT	High, need ()	N/A	High, expressions need parentheses.									
Conversions	**HEX format for integers**	&0-&FFFF	N/A	&0-&FFFF	&0-&FFFFFFFF						&H0-&HFFFFFFFF		&H0-&HFFFFFFFF
					DECI FFFFFFFF		N/A		DECI FFFFFFFF				&FF..., DECI FFFFFFFF
	HEX display	N/A			HEX n, $n<2^{32}$ (display only)		N/A		HEX$(n), $n<2^{32}$, var. l. HEX n (display only)		N/A		HEX$(n), $n<2^{32}$, variable length HEX n (display only, not PC-G850)
	Deg/min/sec to decimal	DEG(d.mmss), DMS values are formatted as a single number											
	Number to deg/min/sec	DMS(d) returns a number formatted as d.mmss											
	Degrees to radians	d / 180 * PI											
	Radians to degrees	r / PI * 180											
	Polar to rectangular	x=r * COS θ, y=r * SIN θ			REC(r,θ) ->Y,Z		Workaround see left		REC(r,θ) -> x=Y, y=Z				
	Rectangular to polar	r=√(x*x + y*y) θ=ACS(x / r)			POL(x,y) -> Y,Z See PC-1280.				POL(x,y) -> r=Y, θ=Z Both functions return variable Y.				
	More conversions	N/A											

Functions

	Vendor	Sharp										
	Model	PC-1500A	PC-1210 PC-1211 PC-1212	PC-1245 PC-1246 PC-1247 PC-1248 PC-1251	PC-1401 PC-1421	PC-1403	PC-1260 PC-1261 PC-1262	PC-1350 PC-1360 PC-2500	PC-1280	PC-1475	PC-E220 PC-G820	PC-G850	PC-E500 PC-E500S
Logs, powers, roots	Common logarithm $\log_{10}x$	LOG x										
	Natural logarithm ln x, $\log_e x$	LN x										
	More accurate ln(x+1)	LN(x + 1)										
	Common antilogarithm 10^x	10 ^ x			TEN x		10 ^ x		TEN x			
	Natural antilogarithm e^x	EXP x										
	More accurate e^{x-1}	EXP x – 1										
	Exponent part of number	INT LOG ABS x for x <> 0										
	Square root √x	SQR x, √x	Symbol √x	SQR x, symbol √x						SQR x		
	Cube root ∛x	x ^ (1/3), x ≥ 0			CUR x		x ^ (1/3), x ≥ 0		CUR x			
	General root $^y\sqrt{x}$	x ^ (1 / y)			x ROT y		x ^ (1 / y)		x ROT y			
	Square x^2	x * x	XX		SQU x		x * x		SQU x			
	Cube x^3	x ^ 3	XXX		CUB x		x ^ 3		CUB x			
Trigonometrics, hyperbolics	sin x, cos x, tan x	SIN x, COS x, TAN x										
	$\sin^{-1}$ x	ASN x										
	$\cos^{-1}$ x	ACS x										
	$\tan^{-1}$ x	ATN x										
	sec x, cosec x, cotan x	1 / COS x, 1 / SIN x, 1 / TAN x										
	Angle to x-axis	ATN(y / x), result probably in wrong quadrant										
	sinh x	(EXP x – EXP(-x)) / 2			HSN x		Workaround see left		HSN x			
	cosh x	(EXP x + EXP(-x)) / 2			HCS x				HCS x			
	cosh x	1–2*EXP x/(EXP x+EXP(-x))			HTN x				HTN x			
	$\sinh^{-1}$ x	LN(x + SQR(x*x + 1))			AHS x				AHS x			
	$\cosh^{-1}$ x	LN(x + SQR(x*x – 1))			AHC x				AHC x			
	$\tanh^{-1}$ x	LN((1 + x) / (1 – x)) / 2			AHT x				AHT x			
	Angle mode degree	DEGREE										
	Angle mode radian	RADIAN										
	Angle mode grad	GRAD										
Probability	Factorial n!	N/A			FACT n		N/A		FACT n			
	Permutations nPr				NPR(n,r)				NPR(n,r)			
	Combinations nCr				NCR(n,r)				NCR(n,r)			
	Random number	RND ctrl	N/A	RND ctrl, 0 < ctrl < 1: 0 ≤ result < 1, ctrl > 2: int result < CEIL(ctrl), ctrl < 0: use previous ctrl value								
	Set random seed	RANDOMIZE		RANDOMIZE								
Statistics	Clear statistics registers	N/A			Only in manual STAT mode		N/A		Interactive application. Japanese prompts on G820 and G850!			
	Add data point											
	Remove data point											
	List sums and results											
	Sums											
	Means and standard deviations											
	Linear regression coefficients											
	Linear estimations											

Functions

Vendor		Sharp										
Model		PC-1500A	PC-1210 PC-1211 PC-1212	PC-1245 PC-1246 PC-1247 PC-1248 PC-1251	PC-1401 PC-1421	PC-1403	PC-1260 PC-1261 PC-1262	PC-1350 PC-1360 PC-2500	PC-1280	PC-1475	PC-E220 PC-G820	PC-G850	PC-E500 PC-E500S
Various functions	Set time and date	TIME= MMDDhh.mmss	N/A										
	Get time and date	TIME											
	Get timer value												
	Read memory	b=PEEK a	N/A	b=PEEK a On some machines you can only PEEK the RAM addresses.									
	Modify memory	POKE a,b1,b2,...		POKE a,b1,b2,... On the low end machines (124x), a colon after the command may be necessary.									
	Get variable address	Fixed addresses		Fixed variables are at fixed addresses.									N/A
	Call machine language	CALL addr,params		CALL address									
				N/A							CALL #bank, address calls ROM bank		N/A
	User defined function	N/A											
	Multi line												
	Recursion												
	Swap Variables	N/A											
	More functions	PC-1421 financial functions: COMP <fin var>, AMRT, ACC, BGNON, BGNOFF, DAYSI(dd.mmyyyy,dd.mmyyyy), DAYSII()											

Functions

	Vendor	HP		TI		Tandy Radio Shack	Canon	Epson
	Model	HP-75	HP-71	TI-74	CC-40	TRS-80 Model 100	X-07	HX-20
	Syntax remarks	All function arguments need parentheses.				All function arguments need parentheses.		
	Precision of mathematical functions	REAL		Default precision		Double precision		Single precision (most)
Strings	Concatenation	string1 & string2				string1 + string2		
Strings	Substrings	String[from,to] <to> defaults to end of string. Can also be on left side of assignment: Substring is cut and replacement inserted.		SEG$(string,start,length)		LEFT$(string,length) RIGHT$(string$,length) MID$(string,start,length)		
Strings	Length	LEN(string)				LEN(string)		
Strings	ASCII to string	CHR$(code)				CHR$(code)		
Strings	String to ASCII	NUM(char)		ASC(char)		ASC(char)		
Strings	Number to string	STR$(expression)				STR$(expression)		
Strings	String to number	VAL(string), string must be valid expression. Numeric expressions are evaluated.		VAL(string), string must be valid number. NUMERIC(string) tests if string is a number. Numeric expressions not supported.		VAL(string), string must be valid number.		
Strings	Expression evaluation					Numeric expressions not supported.		
Strings	Search substring	POS(string,substring)				INSTR(start,string,substring)		
Strings	Case conversion	UPRC$(string)		N/A		N/A		
Strings	Repeat string	N/A		RPT$(string,count)		SPACE$(length), STRING$(length,code or char) only single character is supported.		
Numbers	Absolute Value	ABS(x)				ABS(x)		
Numbers	Sign	SGN(x)				SGN(x)		
Numbers	Integer part	IP(x)		SGN(x) * INT(ABS(x))		FIX(x)		
Numbers	Fractional part	FP(x)		SGN(x) * (ABS(x) – INT(ABS(x)))		x – FIX(x)		
Numbers	Largest integer below or equal	INT(x), FLOOR(x)		INT(x)		INT(x)		
Numbers	Smallest integer above or equal	CEIL(x)		-INT(-x)		-INT(-x)		
Numbers	Round to d decimal places. Examples round to cents. Workarounds for positive x only!	IP(x * 10^d + 0.5) / 10^d, IP(X*100+0.5)/100		INT(x * 10^d + 0.5) / 10^d, INT(X*100+0.5)/100		INT(x * 10^d + 0.5) / 10^d, INT(X*100+0.5)/100		
Numbers		Display only: PRINT USING"DDD.DD";X		Display only: PRINT USING "###.##";X USING works for current line only.		Display only: PRINT USING "###.##";X USING works for current line only.		
Numbers		N/A	FIX d, SCI d, ENG d set display precision					
Numbers	PI	PI				4*ATN(1)		
Numbers						Result is double precision		Result is single precision
Numbers	Other constants	VER$, INF	VER$, INF, NAN, EPS, MINREAL, MAXREAL	N/A	CALL VERSION(V) returns BASIC version.	MAXRAM returns highest available RAM address.	N/A	
Numbers	Maximum	MAX(a,b)		-(a > b) * a - (a <= b) * b		-(a > b) * a - (a <= b) * b		
Numbers	Minimum	MIN(a,b)		-(a > b) * b - (a <= b) * a		-(a > b) * b - (a <= b) * a		
Arithmetic	Implied multiplication AB=A*B	No				No		
Arithmetic	Power x^y	x ^ y				x ^ y		
Arithmetic	Integer division	a DIV b, a \ b		INT(a / b)		a \ b (Yen symbol on X-07)		
Arithmetic	Modulo	MOD(a,b)		a – b * INT(a / b)		a – b * INT(a / b)		
Arithmetic	Remainder	RMD(a,b)		SGN(a) * (ABS(a) – ABS(b) * INT(ABS(a/b)))		a MOD b		
Arithmetic	Reduction	N/A	RED(a,b)	N/A		N/A		
Arithmetic	Percentage	a * p / 100	p % a	a * p / 100		a * p / 100		
Logic	Comparisons	<, <=, =<, >, >=, =>, =, <>, ><, # HP-71 only: ? (unordered, IEEE math)		<, <=, =<, >, >=, =>, =, <>, ><		<, <=, =<, >, >=, =>, =, <>, ><		
Logic	Result of 1=1	1		-1		-1		
Logic	Logical operators	NOT, AND, OR, EXOR	See HP-75.	NOT, AND, OR, XOR		NOT, AND, OR, XOR, EQV, IMP		
Logic	Number of bits	Logical only: nonzero values mapped to 1	The HP-IL module adds BINCMP(n), BINAND(n,m), BINIOR(n,m), BINEOR(n,m) and BIT(n,b) for 20 bit integers.	16, signed		16, signed		
Logic	Priority of NOT	High, expressions need parentheses.		Low		Low		
Conversions	HEX format for integers	N/A	HTD("0")-HTD("FFFFF")	N/A		N/A	&H0-&HFFFF	
Conversions							&nnnnn, &Onnnnn denote octal	
Conversions	HEX display		DTH$(n), n<$2^{20}$, 0-padded to 5 digits.				HEX$(n), n<$2^{16}$	OCT$(n) returns octal string
Conversions	Deg/min/sec to decimal	N/A		Only in CALC mode	N/A	N/A		
Conversions	Number to deg/min/sec							
Conversions	Degrees to radians	RAD(d)		d / 180 * PI		d / 180 * PI (PI must be defined)		
Conversions	Radians to degrees	DEG(r)		r / PI * 180		r / PI * 180 (PI must be defined)		
Conversions	Polar to rectangular	x=r * COS θ, y=r * SIN θ				x=r * COS θ, y=r * SIN θ		
Conversions	Rectangular to polar	r=SQR(x*x + y*y), θ=ACOS(x / r) Keyboard function in TI-74 CALC mode.				r=SQR(x*x + y*y), θ=ACOS(x / r)		
Conversions	More conversions	N/A	In MATH Module	N/A		CINT, CDBL, CSNG convert between integer, double or single precision.		

Functions

	Vendor	HP		TI		Tandy Radio Shack	Canon	Epson
	Model	HP-75	HP-71	TI-74	CC-40	TRS-80 Model 100	X-07	HX-20
Logs, powers, roots	Common logarithm $\log_{10}x$	LOG10(x)	LGT(x), LOG10(x)	LOG(x)		LOG(x) / LOG(10)		
	Natural logarithm ln x, $\log_e x$	LOG(x)	LN(x), LOG(x)	LN(x)		LOG(x)		
	More accurate ln(x+1)	LOG(x + 1)	LOGP1(x)	LN(x + 1)		LN(x + 1)		
	Common antilogarithm 10^x	10 ^ x				10 ^ x		
	Natural antilogarithm e^x	EXP(x)				EXP(x)		
	More accurate e^{x-1}	EXP(x) – 1	EXPM1(x)	EXP(x) – 1		EXP(x) – 1		
	Exponent part of number	INT(LGT(ABS(x)))	EXPONENT(x)	INT(LOG(ABS(x))) for x <> 0		INT(LOG(ABS(x))) for x <> 0		
	Square root √x	SQR(x)	SQR(x), SQRT(x)	SQR(x)		SQR(x)		
	Cube root ∛x	x ^ (1/3), x ≥ 0				x ^ (1/3), x ≥ 0		
	General root $^y\sqrt{x}$	x ^ (1 / y)				x ^ (1 / y)		
	Square x^2	x * x				x * x		
	Cube x^3	x ^ 3				x ^ 3		
Trigonometrics, hyperbolics	sin x, cos x, tan x	SIN(x), COS(x), TAN(x)				SIN(x), COS(x), TAN(x)		
	$\sin^{-1}$ x	ASIN(x)	ASN(x), ASIN(x)	ASN(x)	ASIN(x)	ATN(x / SQR(1 – x*x))		
	$\cos^{-1}$ x	ACOS(x)	ACS(x), ACOS(x)	ACS(x)	ACOS(x)	ATN(x / SQR(1 – x*x))		
	$\tan^{-1}$ x	ATAN(x)	ATN(x), ATAN(x)	ATN(x)		ATN(x)		
	sec x, cosec x, cotan x	SEC(x), CSC(x), COT(x)	1 / COS(x), 1 / SIN(x), 1 / TAN(x)			1 / COS(x), 1 / SIN(x), 1 / TAN(x)		
	Angle to x-axis	ANGLE(x,y)		ATN(y / x), result probably in wrong quadrant		ATN(y / x), result probably in wrong quadrant		
	sinh x	(EXP(x) – EXP(-x)) / 2		SINH(x)	(EXP(x) – EXP(-x)) / 2	(EXP(x) – EXP(-x)) / 2		
	cosh x	(EXP(x) + EXP(-x)) / 2		COSH(x)	(EXP(x) + EXP(-x)) / 2	(EXP(x) + EXP(-x)) / 2		
	cosh x	1–2*EXP(x)/(EXP(x)+EXP(-x))		TANH(x)	1–2*EXP(x)/(EXP(x)+EXP(-x))	1–2*EXP(x)/(EXP(x)+EXP(-x))		
	$\sinh^{-1}$ x	LOG(x + SQR(x*x + 1))		ASINH(x)	LN(x + SQR(x*x + 1))	LOG(x + SQR(x*x + 1))		
	$\cosh^{-1}$ x	LOG(x + SQR(x*x – 1))		ACOSH(x)	LN(x + SQR(x*x – 1))	LOG(x + SQR(x*x – 1))		
	$\tanh^{-1}$ x	LOG((1 + x) / (1 – x)) / 2		ATANH(x)	LN((1 + x) / (1 – x)) / 2	LOG((1 + x) / (1 – x)) / 2		
	Angle mode degree	OPTION ANGLE DEGREES		DEG		All angles are in radians and must be converted in advance.		
	Angle mode radian	OPTION ANGLE RADIANS		RAD				
	Angle mode grad	GRAD is N/A. OPTION ANGLE optional on HP-71		GRAD				
Probability	Factorial n!	N/A	FACT(n)	Only in CALC mode	N/A	N/A		
	Permutations nPr		FACT(n) / FACT(n – r)					
	Combinations nCr		FACT(n) / (FACT(n – r)*FACT(r))					
	Random number	RND			RND, INTRND(bound)	RND(ctrl) ctrl > 0: next in series ctrl = 0: repeat last # ctrl < 0: new series	RND(ctrl) ctrl > 0: next in series ctrl = 0: seed automatically ctrl < 0: seed with ctrl value	RND(ctrl) ctrl > 0: next in series (default) ctrl = 0: repeat last # ctrl < 0: new series
	Set random seed	RANDOMIZE seed If seed is omitted use system value.				FOR I=1 TO VAL(RIGHT$(TIME$,2)): D=RND(1):NEXT		RANDOMIZE seed If seed is omitted user is prompted.
Statistics	Clear statistics registers	N/A	STAT Array(# of cols) up to 15 columns CLSTAT clears current	Only in CALC mode	N/A	N/A		
	Add data point		ADD x1,x2,...					
	Remove data point		DROP x1,x2,...					
	List sums and results		Display the array					
	Sums		TOTAL(0), TOTAL(column)					
	Means and standard deviations		MEAN(col) SDEV(col)					
	Linear regression coefficients		LR col-y, col-x, A, B					
	Linear estimations		PREDEV(x) (after LR)					

Functions

Vendor		HP		TI		Tandy Radio Shack	Canon	Epson
	Model	**HP-75**	**HP-71**	**TI-74**	**CC-40**	**TRS-80 Model 100**	**X-07**	**HX-20**
Various functions	**Set time and date**	Use TIME mode	SETDATE, SETTIME, ADJABS, ADJUST, ...	N/A		TIME$="hh:mm:ss" DATE$="mm/dd/yy", DAY$="xxx" DATE$ format for Europe: "dd/mm/yy".	TIME$="hh:mm:ss" DATE$="yyyy/mm/dd" Unchanged parts may be omitted.	TIME$="hh:mm:ss" DATE$="mm/dd/yy" DAY=d (1..7)
	Get time and date	TIME, DATE, TIME$, DATE$ (YY/MM/DD)				TIME$, DATE$, DAY$	TIME$, DATE$	TIME$, DATE$, DAY
	Get timer value	TIME				N/A		
	Read memory	N/A	PEEK$(adr$,nibbles)	Can be installed	CALL PEEK(address,b1,b2,...)	PEEK(address), INP(port)		PEEK(address)
	Modify memory		POKE adr$,hex$		CALL POKE(address,b1,b2,...)	POKE address,byte, OUT port,byte		POKE address,byte
	Get variable address		N/A	N/A		VARPTR(var)		
	Call machine language	Use external development system and LEX files		Can be installed	CALL EXEC(address,parameters)	CALL address,A,HL	EXEC address	
					CALL GETMEM reserves space.		A=USR(address,param)	DEF USRn=address (n=0..9) A=USRn(param) (n=0..9)
	User defined function	DEF FN X(...)=..., DEF FN X$(...)=... Names follow variable syntax.		SUB/SUBEND define a procedure which can return a value via a parameter. Function return values are not available.		N/A	DEF FN X(...)=..., DEF FN X$(...)=... Names follow variable syntax.	
	Multi line	DEF FN X(...) / LET FN X=... / FN END					N/A	
	Recursion	Allowed		Not allowed				
	Swap Variables	N/A		N/A		SWAP var1,var2 – With string variables, only the pointers are swapped.	N/A	SWAP var1,var2 – With string variables, only the pointers are swapped.
	More functions	N/A		N/A		N/A		

Commands

Vendor		Casio											
	Model	**FX-702**	**PB-100 PB-300 FX-700P FX-710P**	**PB-220 FX-720P**	**FX-730P FX-770P FX-785P FX-790P FX-795P**	**PB-700**	**PB-770**	**FX-750P**	**FX-850P FX-880P**	**VX-4**	**Z-1GR**	**PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B**	**FP-200**
Program flow	**Labels**	N/A									*Label: (>30 chars allowed)	N/A	
	Syntax for branch targets besides line numbers	GTO/GSB instead of GOTO/GOSUB, syntax as with PB-100.	#area (0-9)			PROG area (0-9)			#area (0-9)		#area, *Label	"File"	PROG area (0-9)
			Number or area may be an expression, even in ON ... GOTO targets.						Targets are constant.			"File" may be expression.	Targets are constant.
	ON ... GOTO/GOSUB	GTO variable	GOTO var	Available		GOTO variable		Available					
	IF ... THEN ...	Use ";" before commands, THEN only before jump targets.		THEN is mandatory, ";" is obsolete.		THEN is mandatory			THEN is mandatory except before GOTO				
	IF ... THEN ... ELSE ...	ELSE is N/A.				Available							
	Nested IF	Allowed. Unambiguous because ELSE is not available.				Allowed, nearest ELSE belongs to nested IF.							
	Multiline IF ... ENDIF	N/A									Available	N/A	
	WHILE ... WEND												
	REPEAT ... UNTIL												
	SWITCH ... CASE ... ENDSWITCH												
	FOR I=1 TO 2 STEP -1 NEXT I	Loop is executed once, I=0 after loop.							Loop not executed, I=1 after loop.				
	Variable name on NEXT	Mandatory							Optional, multiple variables allowed.				
	Function and position of END	Executable command anywhere in program. Not executable from keyboard.							Executable command anywhere in program. Closes all files. Executable from keyboard.				
Subroutines	**Local procedure definition besides GOSUB/RETURN**	Use separate program area.									Use program area or set a label.	Use extra file.	Use separate program area.
	Return from procedure	RET	RETURN, Z-1GR supports RETURN target										
	Variable scope	All variables are global.											
	Call and parameter passing	GSB #area	GOSUB #area.			GOSUB PROG area			GOSUB #area.			GOSUB "file"	GOSUB PROG area
	Recursion	Recursion is possible. Local variables must be emulated by arrays.											
Error handling	**ON ERROR**	N/A			MODE 99,1	N/A			ON ERROR GOTO				N/A
	Error line and error code								ERL, ERR				
	Return from error handler				N/A				RESUME, RESUME NEXT, RESUME target				
	Disable error handler				MODE 99,0				ON ERROR GOTO 0				
	More event handling				N/A				N/A			A file named "AUTO.EXE" is executed on power up.	
	Debugging	MODE 2, MODE 3 turn tracing on/off.				TRON, TROFF turn tracing on/off.							
	Suspend execution	STOP											
	Continue after STOP, break key or break point	CONT key	EXE key on empty input line.			CONT			EXE key on empty input line.			CONT key OM-53B: Shift+Down	STOP/CONT key

Commands

	Vendor	Casio											
	Model	FX-702	PB-100 PB-300 FX-700P FX-710P	PB-220 FX-720P	FX-730P FX-770P FX-785P FX-790P FX-795P	PB-700	PB-770	FX-750P	FX-850P FX-880P	VX-4	Z-1GR	PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B	FP-200
Display	Clear display	PRT without items	PRINT without items			CLS							
	Output to display	PRT, DMS var	PRINT										
	Behavior of comma	Clear display before output, pause				New line		Pause and new line		New line			Tabulate (12 chars).
	PRINT ends with ; or ,	Allowed											Allowed. ";" between items is optional.
	Default display mode	Line by line				Continuous output		Line by line		Continuous output			
	Continue after PRINT	CONT key	EXE key					ENTER/RET key	EXE key				
	Position cursor	PRT CSR n	PRINT CSR n			PRINT TAB(n)		TAB(n) only with LPRINT	PRINT TAB(n)				
						LOCATE x,y		LOCATE x	LOCATE x,y Virtual screen				LOCATE x,y
	Set display delay	N/A						WAIT n WAIT 999 (off) Unit is 0.05s.	N/A				
	Display formatting	SET Fd, SET Ed, SET N (re)set display precision for numbers.				PRINT USING "&&&&";A$;B$;... PRINT USING "###.##^";X,Y Works for current line only. Only one format per format string allowed.			SET Fd, SET Ed, SET N (re)set display precision			PRINT USING"& &#,###.##";A$;X Works on current PRINT/LPRINT statement only. Mixed formats are allowed. "!" outputs single char, "@" formats a string with its exact length. "##.##^^^^" sets scientific notation.	
												N/A	"+" or "-" can be prefix or postfix, "$$", "**" and "**$" pad numbers to the left.
	Reverse (light on dark)	N/A									PRINT REV;...;NORM;...		N/A
	Graphics screen	N/A				160x32		N/A		192x32			159x63 with arbitrary scaling.
	Query dot or pattern					POINT(x,y)				Commands not implemented	POINT(x,y)		
	Set/reset dot					DRAW/DRAWC(x,y)					DRAW/DRAWC(x,y)		
	Draw (filled) rectangle					DRAW[C](x1,y1)-(x2,y2)-...-(x1,y1) Multiple points, no filling					LINE(..)-(..), mode,BF mode=0: clear mode=1: draw F fills	Use DRAW, no filling.	QUAD(..)-(..) QUADC(..)-(..) No filling.
	Draw line or polygon										DRAW[C](x1,y1)-(x2,y2)-...-(x1,y1) Multiple points		
	Graphical patterns					N/A				DEFCHR$(c)="hex(12)" Defines char c ≥ 252 (c ≥ 240 on PB-1000).			N/A
	More graphics commands									N/A	CIRCLE(x,y), r,mode PAINT(x,y)	N/A	INIT(X0,Y0),DX,DY sets origin and number of pixels for a unit of 1.
	(x,y) outside screen area					Error					Error		
Printer	Printer interface and type	FP-10 (matrix)	FA-3 with FP-12S (matrix)			FA-10 or FA-4 with FP-100 (pen plotter)		FA-20 (matrix)	FP-100 (pen plotter) connected through FA-6 (except PB-1000, which uses FA-7 or MD-100)				Centronics.
	Printer output	LIST/PRT in MODE 7	LIST/PRINT in MODE 7			LLIST/LPRINT		LLIST/LPRINT or redirected LIST/PRINT				LLIST/LPRINT	
	Redirect display to printer	MODE 7/8 turn printer on/off				N/A		PRINT ON/OFF	MODE 7/8 turn printer on/off			N/A	
	Set width for printer output	N/A											
	Set Printer to text or graphics mode	N/A				LPRINT CHR$(28);CHR$(x) x=46: text, x=37: graphics (FA-10 plotter)		N/A				See PB-700	N/A
	Printer commands in graphics mode					Use LPRINT to send plotter commands.							
	Additional printer commands in text mode					N/A						LPRINT escape sequence	

Commands

Vendor		Casio										
	Model	**FX-702**	**PB-100 PB-300 FX-700P FX-710P**	**PB-220 FX-720P**	**FX-730P FX-770P FX-785P FX-790P FX-795P**	**PB-700**	**PB-770**	**FX-750P**	**FX-850P FX-880P**	**VX-4**	**Z-1GR**	**PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B**	**FP-200**
Sound	**Beeper**	N/A		BEEP pitch									N/A
	Frequency range			0: low pitch (default), 1: high pitch									
Input	**Interactive data input**	INP"prompt",variable(s), "prompt",variables(s),...	INPUT"prompt",variable(s),"prompt",variable(s),...		INPUT"prompt",variable(s),"prompt";variable(s),...								INPUT "prompt";variable(s) Only one prompt allowed.
	Behavior of comma or semicolon after prompt	Semicolon is not allowed. Prompt is always followed by "? " and display is cleared upon first key press.			Semicolon adds "?" to prompt, comma suppresses it.								Comma suppresses "?" after prompt.
	Allowed input values and keys	Numeric expression or unquoted string must be entered. You cannot skip input. Values are searated by EXE (resp. ENTER/RETURN)							Numeric expression or unquoted string. Values are separated by EXE. Empty input leaves numbers unchanged and sets strings to an empty string.				Number or (quoted) string. Unquoted strings are stripped on both ends.
		EXE without value suspends execution, CONT resumes. AC clears error condition.	STOP suspends execution, EXE resumes input. AC clears error condition. IN aborts program.		CLS clears screen but input continues. BRK aborts (CONT does not work.) ANS/STOP works as answer key.				Arrow keys allow full screen edit. Rest of line from cursor is accepted as input.				Values are separated by commas. Empty input causes ?TM error except for a single string variable.
	Read keyboard directly	A$=KEY, returns "" if no key pressed.		A$=KEY$, returns "" if no key pressed.	A$=INKEY$, returns "" if no key pressed.								
					A$=INPUT$(count) returns exactly count key presses.								N/A
	Some special key codes	Only unshifted codes and no special keys returned by KEY or KEY$			EXE=13, LEFT/RIGHT=29/28, UP/DOWN=30/31, FX-850P/880P only: S=14, no combined codes PB-1000: Display sensor fields return 240..255 (top left to bottom right), only with INKEY$ not INPUT$()								ENTER=13, LEFT/RIGHT=29/28, UP/DOWN=30/31, DEL=17, HOME=11 PF-Keys return strings.
	Read display contents as input	ANS key returns last result.	N/A		ANS key returns last result.				Full screen editor works in INPUT mode.				N/A
	DATA/READ/RESTORE	N/A		Data elements are quoted or unquoted strings or numeric constants. The line number in RESTORE may be an expression.									Data elements are quoted or unquoted strings or numeric constants.
Program files	**Tape filename syntax**	"NNNN..." (up to 8 identical chars) Name can be omitted and defaults to unnamed or first file found.	"name" (8 chars) Name can be omitted and defaults to unnamed or first file found.						"CASp:(s)name" p=0/1 (phase) s=S/F (slow/fast) Default is "CAS0:(F)" Name is 8 characters. All parts of name can be omitted.	No tape interface commands.	"CAS0:name" Name is 8 chars. Phase & speed are set with switches. At least "CAS0:" must be provided.		"CAS0:name" Name is 8 chars and can be omitted.
	Other storage devices	N/A							Serial I/O: "COM0:parameters" Floppy disk: "0:name8.ext" (No floppy with FX-850P/FX-880P)		Serial, floppy see left. RAM disk "name8.ext"		Serial I/O: "COM0:" (baud rate is 300 fix, 7 bits, even parity) Floppy disk: "0:name8.ext"
	Save program to tape in binary	SAVE #area "name"	SAVE "name"	SAVE "name"	SAVE "name"					N/A	SAVE "CAS0:name"		SAVE "CAS0:name" Cassette is default device.
	Save multiple programs	SAVE ALL "name"	SAVE A "name"	SAVE ALL "name" (tape only)							N/A		
	Set (password) protection	Password must be set beforehand with PASS "password" and will be recorded with SAVE and SAVE ALL. Password protection disables ASCII save (e. g. to serial I/O). Casio tape utilities can reveal the password.											PASS "password" sets password for all areas and inhibits SAVE or LIST. SAVE "name","password" sets password for cassette file.
		Password is active for complete machine.									Password is active for current file only.		
	Save program to other device	N/A							SAVE "name" (Serial I/O switches to ASCII.) Z-1GR file system only accessible from F.COM menu.				SAVE "name"
	Save in ASCII format	Use list730 utility on PC to convert binary tape file.			SAVE "name",A (tape only) Use list850 utility on PC to convert tape file.				SAVE "name",A (Serial I/O defaults to ASCII.) Use list850 utility on PC (not for Z-1GR).				SAVE "name",A
	Load binary program from tape	LOAD #area "name" Programmable command.	LOAD "name"							N/A	LOAD "CAS0:name"		LOAD "CAS0:name"
	Load multiple programs	LOAD ALL "name"	LOAD A "name"	LOAD A/ALL "name"	LOAD ALL "name"						N/A		
	Load binary program from storage	N/A							LOAD "name" (Serial I/O reads ASCII only.) Z-1GR file system only accessible from F.COM menu.				LOAD "name" Format is detected.
	Load ASCII program	Use bas730 utility on PC to create binary tape file from source.			LOAD "name",A (tape only). Use bas850 utility on PC to create tape file.				LOAD "name",A (Serial I/O defaults to ASCII). Use bas850 utility on PC to create tape file.		LOAD "name" Format is detected.		
	Load "foreign" program				Use ASCII format or bas850 on PC.				PBLOAD phase "name" (not VX) Use slow ASCII tape format, serial I/O or PC.	Only via serial I/O.	Use slow ASCII tape format, serial I/O or PC.		Use serial I/O in ASCII format.

Commands

Vendor		Casio										
	Model	**FX-702**	**PB-100 PB-300 FX-700P FX-710P**	**PB-220 FX-720P**	**FX-730P FX-770P FX-785P FX-790P FX-795P**	**PB-700**	**PB-770**	**FX-750P**	**FX-850P FX-880P**	**VX-4**	**Z-1GR**	**PB-1000 PB-2000C/AI-1000 with BASIC ROM OM-53B**	**FP-200**
Program files	**MERGE program lines**	All perfomed in one go by LOAD. Exisiting program is cleared from first line of loaded file, which is automatically started if LOAD is executed from a running program.	Like FX-702P but without automatic execution	N/A						MERGE "name"			N/A
	Handling of duplicate line numbers									Lines are replaced, file type must be ASCII.			
	Run program from storage or tape		N/A			CHAIN "name"			N/A	CHAIN "name"		CHAIN "name" GOTO "name" (RAM disk)	LOAD "name",R
	SAVE or LOAD special areas	N/A		SAVE#"name" saves MEMO. LOAD#"name",M loads/merges MEMO.		N/A			SAVE#/LOAD# of MEMO, see left.	Use F.COM menu for ASCII areas. Z1-GR only save/load memory: BSAVE start,end BLOAD start,end,R		Use menu to copy any type of file to any device.	CETL has interactive G and P commands.
	Check integrity of file	VER "name" performs checksum test on tape file.		VERIFY "name" performs checksum tests on tape file									
	Rename file	No file system								NAME "source" AS "destination"		Use menu.	N/A
	Delete file									KILL "file"			KILL "0:file"
	Copy file									Use F.COM menu			N/A
	List directory									FILES "pattern"			FILES "0:name"
	Format storage medium									FORMAT /cap, cap=6/9/M		FORMAT	FORMAT "0:"
Data files	**OPEN channel on device or file**	MEMO N/A		MEMO data base can be accessed like a single RAM file.		N/A			OPEN "name" FOR mode AS #channel				
	Valid OPEN modes and channels								INPUT/OUTPUT, #1	INPUT/OUTPUT/APPEND, #1..16			INPUT/OUTPUT, #1..#16 The maximum number of file descriptors must be specified with MOUNT <number>. memory is taken from CETL area.
	Close channel								CLOSE	CLOSE closes all channels, CLOSE #channel closes a channel.			CLOSE #channel,... CLOSE without channel closes all open files.
	Write data sequentially	PUT "name" first variable,last variable Variables are ordered $, A..T9				PUT "name" var1,var2,... Every single variable must be named.			PRINT #channel, print item; print item; ... Commas, TAB(n) and USING (not FX-850P/880P) allowed. MEMO access with WRITE# (except PB-1000, FP-200)				
		MEMO N/A		WRITE# var1,var2,... Writes to MEMO									
	Read data sequentially	GET "name" first variable,last variable				GET "name" var1,var2,... Every single variable must be named.			INPUT #channel, var1, var2, ...				
		MEMO N/A		READ# var1,var2,... reads from MEMO					var$=INPUT$(count,#channel) LINE INPUT #channel, var$ (Not FX-850P/880P) MEMO access with READ# (except PB-1000)				N/A
	Random access files			RESTORE# string,n,target positions MEMO pointer. n=0: rec starts with string (def) n=1: record contains striing GOTO target if not found.		N/A			MEMO access see left	RESTORE#("Fa") string,n,target Selects ASCII area a and positions MEMO pointer in it. (See left)		OPEN "0:name" AS #chan FIELD #chan,@,len AS var$,... (Record size is 256 bytes) LSET/RSET var$=string PUT/GET #chan,record > 0	OPEN "0:name" AS #chan FIELD #chan,len AS var$,... (Record size is 256 bytes) LSET/RSET var$=string CVD/CVS/MKD$/MKS$ PUT/GET #chan,record > 0
	Special I/O functions	N/A							EOF(channel) tests for end of file.			EOF(channel) tests end of file. LOF(channel) returns length of file or chars left in input buffer.	EOF(channel) tests end of file. LOC(channel) returns next record number. LOF(channel) returns number of records.

Commands

	Vendor	Sharp											
	Model	PC-1500A	PC-1210 PC-1211 PC-1212	PC-1245 PC-1246 PC-1247 PC-1248 PC-1251	PC-1401 PC-1421	PC-1403	PC-1260 PC-1261 PC-1262	PC-1350 PC-2500	PC-1360	PC-1280	PC-1475	PC-E220 PC-G820	PC-G850	PC-E500 PC-E500S
Program flow	Labels	Alphanumeric(7) with " ". Optional colon. "A", "S", "D", "F", "G", "H", "J", "K", "L", "Z", "X", "C", "V", "B", "N", "M" + some more are reachable with DEF key (SHFT in DEF mode on PC-121x).										Alphanumeric(20) with " " or preceded by *. Optional colon.		
	Syntax for branch targets besides line numbers	"Label"										*Label or "Label"		
		Line number or label may be expression.										Targets are constant.		
	ON ... GOTO/GOSUB	Available	GOTO expression	Available. Target expressions must not contain commas.								Available.		
	IF ... THEN ...	THEN is optional. PC-121x allows THEN only as an alias for GOTO with a jump target. If the first statement is an assignment, LET must be used.												
	IF ... THEN ... ELSE ...	ELSE is N/A.											Available.	
	Nested IF	Allowed. Unambiguous because ELSE is not available.											Allowed, nearest ELSE belongs to nested IF.	
	Multiline IF ... ENDIF WHILE ... WEND REPEAT ... UNTIL SWITCH ... CASE ... ENDSWITCH	N/A											Available.	PC-E500S only
	FOR I=1 TO 2 STEP -1 NEXT I	Loop executed once, I=0 after loop.	Loop executed, I=1 (!) Limit and step integer in the range +/-1000.	Loop executed once, I=0 after loop.								Loop not executed, I=1 after loop.		
	Variable name on NEXT	Mandatory										Optional		
	Function and position of END	Executable command anywhere in program. Not executable from keyboard.												
Subroutines	Local procedure definition besides GOSUB/RETURN	Set a label.												
	Return from procedure	RETURN												
	Variable scope	All variables are global.												
	Call and parameter passing	GOSUB "Label"										GOSUB *Label		
	Recursion	Recursion is possible. Local variables must be emulated by arrays.												
Error handling	ON ERROR	N/A												ON ERROR GOTO
	Error line and error code													ERL, ERN
	Return from error handler													RESUME, RESUME NEXT, RESUME target
	Disable error handler													ON ERROR GOTO 0
	More event handling	N/A							ARUN, AUTOGOTO Power on auto start		N/A			ARUN, AUTOGOTO Power on auto start
	Debugging	TRON, TROFF	DEBUG starts in trace mode	TRON, TROFF turn tracing on/off.										
	Suspend execution	STOP												
	Continue after STOP, break key or break point	CONT												

Commands

	Vendor	Sharp											
	Model	PC-1500A	PC-1210 PC-1211 PC-1212	PC-1245 PC-1246 PC-1247 PC-1248 PC-1251	PC-1401 PC-1421	PC-1403	PC-1260 PC-1261 PC-1262	PC-1350 PC-2500	PC-1360	PC-1280	PC-1475	PC-E220 PC-G820	PC-G850	PC-E500 PC-E500S
Display	Clear display	CLS	N/A. Display goes blank while running.				CLS						
	Output to display	PRINT or PAUSE, PC-121x is limited in what it can display after a semicolon.										PRINT		
	Behavior of comma	Split display, only two items allowed.					Tabulate (12 chars), number of items depends on display size.							
	PRINT ends with ; or ,	Allowed	Syntax error		Only semicolon allowed									Allowed
	Default display mode	Line by line	Line by line, display goes blank while running.				Line by line, but scrolling.					Continuous output		
	Continue after PRINT	Enter key												
	Position cursor	CURSOR n	N/A				CURSOR n On multi line screen n>display length is 2nd line, and so on.					LOCATE x,y		
		GCURSOR x Unit is pixel	N/A				See PC-1360. Needs GOTO between CURSOR and INPUT and crashes if INPUT prompt ends on last screen position.	CURSOR x,y CURSOR without arguments releases position for PRINT after CURSOR positioning followed by INPUT.						
	Set display delay	WAIT see right Use PAUSE instead of PRINT, delay is fixed at 0.85s.	N/A	WAIT n, unit is 0.01562s (1/64), maximum 65535. WAIT without argument sets infinite wait.								N/A		
	Display formatting	USING "###.##^" sets number display format, USING "&&&" sets string display format. USING can be used within PRINT/PAUSE/LPRINT or standalone. USING without format resets the formatting. USING is persistent, even if used together with PRINT (except on PC-E500). Separate formats are kept for strings and numbers. PC-122x does not support string formats. All later models support mixed formats (but without constant strings). PC-E500 supports mixed formats with constant strings like "&&&/##.#%".												
	Reverse (light on dark)	N/A	N/A				LCD is memory mapped. See special commands page for details.	See G850	N/A				Emulate with LINE(..)-(..),X,BF	
	Graphics screen	156x7						150x32					144x48	240x32
	Query dot or pattern	pattern=POINT x						POINT(x,y)					POINT(x,y)	
	Set/reset dot	GCURSOR x positions cursor GPRINT "hex" outputs bits. Can be freely mixed with PRINT.						P[RE]SET(x,y),X/S/R					P[RE]SET(x,y),X/S/R	
	Draw (filled) rectangle							LINE(..)-(..), X/S/R,BF F fills box					LINE(..)-(..), X/S/R,pattern,BF F fills box	
	Draw line or polygon							LINE(..)-(..),X/S/R Continues if started with -					LINE(..)-(..),X/S/R,pattern Continues if started with -	
	Graphical patterns							GCURSOSR x,y GPRINT pattern;... Hexstrings or numbers					GCURSOSR x,y GPRINT pattern;... Hexstrings or numbers	
	More graphics commands							N/A					N/A	
	(x,y) outside screen area	Error						Virtual screen					Virtual screen with clipping	
Printer	Printer interface and type	CE-150 (pen plotter, contains ROM with BASIC commands)	CE-122 (matrix)	CE-126P (matrix), works with all Sharps with 11 pin connector.										
				No other printer available				PC-1350: CE-515P (see right) PC-2500: Built-in pen plotter, no 11 Pin interface.	CE-140P (matrix plotter) CE-515P (pen plotter) Serial printer with level shifter	N/A	See PC-1360	Any serial printer through RS-232 with level shifter.		CE-515P (pen plotter) with CE-516L. Any serial printer through RS-232 with level shifter.
	Printer output	LLIST/LPRINT	Switch on CE-122. Disables some LIST functionality.	LLIST/LPRINT										
	Redirect display to printer	N/A		PRINT=LPRINT turns printer on, PRINT=PRINT turns printer off. Keyboard function P<->NP (switch on PC-2500) switches between these modes.										
	Set width for printer output	N/A						CONSOLE n			CONSOLE n			CONSOLE n
	Set Printer to text or graphics mode	GRAPH TEXT (CE-150)	N/A					PC-2500 only: LPRINT CHR$ &1B;"a"; LPRINT CHR$ &1B;"b";	GRAPH, LTEXT (CE-140P, CE-515P, needs OPEN/CLOSE)	N/A	GRAPH, LTEXT (CE-140P, CE-515P, needs OPEN/CLOSE)	N/A		See PC-1475. Only CE-515P supported. OPEN and CLOSE are mandatory.
	Printer commands in graphics mode	COLOR, CSIZE, GLCURSOR, LINE, RLINE, ROTATE, SORIGIN						PC-2500: Use LPRINT to send plotter commands. TEST sends test pattern	CIRCLE, COLOR, CROTATE, CSIZE, GLCURSOR, LLINE, PAINT, SORGN		CIRCLE, COLOR, CROTATE, CSIZE, GLCURSOR, LLINE, PAINT, SORGN			
	Additional printer commands in text mode	LF, LCURSOR, TAB, TEST						OPEN/CLOSE for CE-515P	LF, OPEN/CLOSE for CE-515P		LF, OPEN/CLOSE for CE-515P			

Commands

	Vendor	Sharp												
	Model	PC-1500A	PC-1210 PC-1211 PC-1212	PC-1245 PC-1246 PC-1247 PC-1248 PC-1251	PC-1401 PC-1421	PC-1403	PC-1260 PC-1261 PC-1262	PC-1350 PC-2500	PC-1360	PC-1280	PC-1475	PC-E220 PC-G820	PC-G850	PC-E500 PC-E500S
Sound	Beeper	BEEP count,frq,time or ON/OFF Duration depends on time and frq.	BEEP count						BEEP count,frq,time Duration depends on time and frq.					
Sound	Frequency range	Frequency ≈ 1.3E6/(166+22*frq) 0: 7kHz, 129: 440Hz, 255: 230Hz	Only a single pitch available. PC-1246 is mute.						Same as PC-E500 My PC-1280 seems to have a speed up installed and beeps higher.			Syntax like PC-1500. Buzzer is not connected on G820 and G850 models.		Frequency = 256000/(90+4*frq) 0: 2844,4Hz, 123: 440Hz, 255: 230,6Hz
Input	Interactive data input	INPUT"prompt",variable(s), "prompt";variables(s):more statements										INPUT"prompt",variable(s), "prompt";variables(s)		
Input	Behavior of comma or semicolon after prompt	Comma forces prompt to be cleared on first key press, semicolon leaves prompt in display. "?" is only displayed if no prompt is specified or after input errors.												
Input	Allowed input values and keys	Numeric expression or unquoted string. Values are separated by ENTER. Empty input lines leave numbers and strings unchanged and skip the rest of the program line. INPUT acts as a conditional statement!										Numeric expression or unquoted string. Values are separated by ENTER. Empty input lines leave numbers and strings unchanged.		
Input	Allowed input values and keys	CL/CLS/C-CE clears input field or error condition. DOWN single steps. UP suspends execution. Shift-CA stops program. BRK/BREAK interrupt programs (can be resumed with CONT). On some machines INPUT has some display problems with multiple prompts or in error situations if the prompt is not positioned in first column.												
Input	Read keyboard directly	A$=INKEY$, returns code repeatedly as long as key is down.	N/A	A$=INKEY$, returns "" if no key pressed.										
Input	Read keyboard directly			N/A										A$=INPUT$(count) returns exactly count key presses.
Input	Some special key codes	ENTER=13, UP/DOWN=11/10, LEFT/RIGHT=8/12, F-Keys=17..22, SHIFT=1, SML=2, no combined codes.		ENTER and other special keys do not return codes. PC-2500 only: ENTER=13, UP/DOWN=30/31, LEFT/RIGHT=29/28, SHIFT-LEFT/RIGHT=2/6, DEL=127, BS=8, INS=18, CLS=12. SHIFT and CAPS do work normally.					ENTER=13, UP/DOWN=4/5, LEFT/RIGHT=15/14, 2nd/SHIFT=16, no combined codes. Most special keys return codes below 32 or above 127.					See left for INKEY$. INPUT$ returns CHR$(00)+code for some special keys (see manual).
Input	Read display contents as input	AREAD var Must be first statement directly after a label that can be reached with the DEF key. Program must be started with DEF+label, ignored otherwise.										N/A		
Input	DATA/READ/RESTORE	See right	N/A	Data elements are quoted strings, string expressions or numeric expressions. The line number in RESTORE may be an expression.								Data elements are quoted or unquoted strings or numeric constants. Restore target may be *label.		
Program files	Tape filename syntax	"name" (16 chars). -1 can be appended to the command name to specify secondary remote jack. Name can be omitted.	"name" (7 chars). Name can be omitted and defaults to unnamed or first file found.									"name" (8 chars) with CSAVE, CLOAD, etc. "CAS:name" with OPEN. Name part can be omitted and defaults to unnamed or first file found		"name" (8 chars) with CSAVE, CLOAD, etc. "CAS:name" with SAVE, LOAD, MERGE, CHAIN, OPEN. Name can be omitted and defaults to unnamed or first file found
Program files	Other storage devices	N/A				Pocket disk: "X:name8.ext"	N/A	Serial I/O: "parameters" PC-1360 only: Pocket disk: "X:name8.ext"		RAM/Pocket disk: "d:name8.ext" d=F/X	Serial I/O: "parameters" RAM/Pocket disk: "d:name8.ext" d=F/X	Serial I/O: "COM:" (not with LOAD/SAVE, OPEN only) RAM disk: "name8.ext" (not with OPEN)		Serial I/O: "COM:parameters" RAM/Pocket disk: "d:name8.ext" d=E/F/X Name defaults to serial I/O if omitted.
Program files	Save program to tape in binary	CSAVE "Name"	CSAVE "name" Format is comaptible.						CSAVE "name" CSAVE@ "name" saves in older format.			CSAVE "name"		CSAVE "name"
Program files	Save multiple programs	N/A						No RAM disk or multiple areas		COPY source TO destination with wildcards from RAM disk to pocket disk.		N/A		COPY source TO destination with wildcards from RAM disk to pocket disk.
Program files	Set (password) protection	CSAVE "Name","Password". Saving is disabled if password is set in memory.	N/A	CSAVE "Name","Password". Saving is disabled if password is set in memory. Compatibility see CSAVE.										
Program files	Set (password) protection			No file system	See PC-1360	N/A			SET "file pattern","P" Makes file(s) readonly. " " removes the protection.			N/A		SET "file pattern","P"/" " Sets/removes write protection.
Program files	Save program to other device	N/A							Pocket or RAM disk: SAVE "d:name" Serial I/O (not PC-1280): OPEN followed by SAVE.			SAVE "name" RAM disk only.		SAVE "name"
Program files	Save in ASCII format								SAVE "d:name",A (Pocket disk only) SAVE to serial I/O (not PC-1280) is always in ASCII			Use TEXT menu. Only way to save to serial I/O.		SAVE "name",A SAVE to serial I/O
Program files	Load binary program from tape	CLOAD "name"	CLOAD "name" Format is compatible: Newer models can read older tapes.									CLOAD "name"		CLOAD "name"
Program files	Load multiple programs	N/A						No RAM disk or multiple areas		COPY source TO destination with wildcards from pocket disk to RAM disk.		N/A		COPY source TO destination with wildcards from pocket disk to RAM disk.
Program files	Load binary program from storage	N/A			See PC-1360	N/A			Pocket or RAM disk: LOAD "name",R R starts program. Format is detected. Serial I/O (not PC-1280): OPEN followed by LOAD.			LOAD "name" RAM disk only.		LOAD "name",R R starts program. Format is detected.
Program files	Load ASCII program	Use BAS2IMG on PC. Link (PC-1500)	N/A	Use BAS2IMG on PC. Link (PC-12xx/13xx/14x								Use TEXT menu. Only way to read from serial I/O.		
Program files	Load "foreign" program								Use ASCII mode, serial I/O or BAS2IMG on PC.	Use ASCII mode or BAS2IMG on PC.	Use ASCII mode, serial I/O or BAS2IMG on PC.	Use TEXT menu, ASCII mode, serial I/O		CLOAD@ "name" CSAVE@ "name" saves in older format.

Commands

	Vendor	Sharp											
	Model	PC-1500A	PC-1210 PC-1211 PC-1212	PC-1245 PC-1246 PC-1247 PC-1248 PC-1251	PC-1401 PC-1421	PC-1403	PC-1260 PC-1261 PC-1262	PC-1350 PC-2500	PC-1360	PC-1280	PC-1475	PC-E220 PC-G820	PC-G850	PC-E500 PC-E500S
Program files	MERGE program lines	MERGE "name"	CLOAD 1 "name" ROM dependent.	MERGE "name"		MERGE "name"						N/A		MERGE "CAS:name" ASCII only
	Handling of duplicate line numbers	Program is appended, duplicates are allowed. Only last copy editable or reachable with GOTO. Use labels!				Program is appended, duplicates are allowed. Only last copy editable or reachable with GOTO. Use labels!								Lines are replaced.
	Run program from storage or tape	CHAIN "name",start The start parameter follows GOTO syntax.				CHAIN "name",start The start parameter follows GOTO syntax.								CHAIN "CAS:name" LOAD with option R
	SAVE or LOAD special areas	N/A		Load or save memory to tape: CSAVEM"name";start,end CLOADM"name";start								Save/load memory in MONitor: Wstart,end Rstart Device is serial I/O.		N/A
		N/A					In RSV mode, CSAVE and CLOAD handle RSV memory.							
	Check integrity of file	CLOAD? "name" (tape only)												CLOAD? "name" (tape)
														LOAD? "name"
	Rename file	No file system			See PC-1360	N/A			NAME source AS destination			N/A		NAME source AS destination
	Delete file								KILL file					
	Copy file								COPY source TO destination			N/A		COPY source TO destination
	List directory								FILES/LFILES pattern					
	Format storage medium								INIT "d:"			N/A		INIT "d:"
Data files	OPEN channel on device or file	N/A			See PC-1360	N/A		OPEN "parameters" opens serial I/O. OPEN$ returns active COM settings.	OPEN "d:name" FOR mode AS #channel OPEN "parameters" opens serial I/O on #1. OPEN$ returns active COM settings.			OPEN "CAS:name" FOR mode OPEN "COM:" Serial parameters set in menu		OPEN "name" FOR mode AS #channel OPEN "parameters" AS #ch opens serial I/O COM$ returns active COM settings.
	Valid OPEN modes and channels							#1 is the only available (serial) channel.	INPUT/OUTPUT/APPEND Serial I/O: #1, Disk: #2..7, RAM: #20..25			INPUT/OUTPUT, #1		INPUT/OUTPUT/APPEND, #1..255 (max 6+2) Any device can take any channel. Serial defaults to #1 if not specified.
	Close channel							CLOSE	CLOSE closes all channels CLOSE #ch1,#ch2,... closes selected channels.			CLOSE #1		CLOSE closes all channels. CLOSE #ch1,#ch2,... selected channels.
	Write data sequentially	PRINT#,"name";var1,var2,... A(*) specifies an array. Name is mandatory if var1 is a string. Every single variable must be named.	PRINT#"name";first Write all variables beginning with first (default A)	See left. Arrays A(*) can be appended to list or specified alone.	PRINT#"name";var1,var2,... (tape only) Items are single variables, Arrays X(*) or fixed variables A* as block start.							Old tape syntax is N/A		
					N/A	See PC-1360	N/A	PRINT#channel,item1,item2,... Items are single variables, string literals or arrays A(*). ";" can replace "," except between strings, which would be merged. Channel must be open for OUTPUT or APPPEND. LPRINT after OPEN works like PRINT#1 (serial I/O).						
	Read data sequentially	INPUT#,"name";var1,var2,... A(*) specifies an array. Name is mandatory if var1 is a string. Every single variable must be named.	INPUT#"name";first Read all variables beginning with first (default A)		INPUT#"name";var1,var2,... (tape only) Variables are single items, Arrays X(*) or fixed variables A* as block start.							N/A		var$=INPUT$(count,#channel)
					N/A	See PC-1360	N/A	INPUT#channel,var1,var2,... Variables are single items or arrays A(*). Channel must be open for INPUT.						
	Random access files	N/A												
	Special I/O functions	N/A							EOF(channel) tests for end of file. LOF(channel) returns length of file or chars left in input buffer. LOC(channel) returns current record (256 bytes long). DSKF(d) returns space on disk: 1: pocket, 3: RAM disk.			N/A		EOF/LOF/LOC/DSKF see left. Parameter d for DSKF is 1 (pocket disk), 3 (E:) or 4 (F:)

Commands

Vendor		HP		TI	
Model		HP-75	HP-71B with HP-IL	TI-74	CC-40
Program flow	Labels	N/A	Alphanumeric(8) with ' ': Same syntax as file names.	N/A	
	Syntax for branch targets besides line numbers		'LABEL', LABEL		
	ON ... GOTO/GOSUB	Available			
	IF ... THEN ...	THEN is mandatory.			
	IF ... THEN ... ELSE ...	Available.			
	Nested IF	N/A	Only after ELSE.	Allowed, nearest ELSE belongs to nested IF.	
	Multiline IF ... ENDIF	N/A			
	WHILE ... WEND				
	REPEAT ... UNTIL				
	SWITCH ... CASE ... ENDSWITCH				
	FOR I=1 TO 2 STEP -1 NEXT I	Loop not executed, I=1 after loop.			
	Variable name on NEXT	Mandatory			
	Function and position of END	Executable command anywhere in program. Closes all local files and deallocates local variables. Substitutes END SUB in subroutine. Returns from CALLed external program. HP-75 : not executable from keyboard.		Executable command anywhere in program. Closes all open files. Allowed even after SUBEND. Executable from keyboard.	
Subroutines	Local procedure definition besides GOSUB/RETURN	Use DEF FN or external file.	SUB name(params) Name follows label syntax.	SUB name(params) Name follows variable syntax.	
	Return from procedure	END, END FN	END SUB, END or next SUB.	SUBEND. SUBEXIT returns early.	
	Variable scope	Parameters of FNX() and variables in external program are local.	Variables are local. Files are local, if no parameter list is defined.	All variables are local.	All variables are local. ATTACH/RELEASE name,... handle preallocation and allow variable persistence.
	Call and parameter passing	X=FNY(...) CALL 'file'	CALL name(R,A,(V),#C) CALL file	CALL name(Reference,Array(),Matrix(,),(Value))	
	Recursion	Fully implemented.		N/A	
Error handling	ON ERROR	ON ERROR command	ON ERROR GOTO/GOSUB	ON ERROR line number	
	Error line and error code	ERRL, ERRN	ERRL, ERRN, ERRM$	CALL ERR(CODE,TYPE,FILE,LINE)	
	Return from error handler	RETURN if command is GOSUB.		RETURN, RETURN NEXT, RETURN line number	
	Disable error handler	OFF ERROR		ON ERROR STOP	
	More event handling	ON TIMER #n,seconds,commands OFF TIMER #n		ON WARNING PRINT/NEXT/ERROR ON BREAK STOP/NEXT/ERROR	
		N/A	DEFAULT ON/OFF/EXTEND, TRAP handle math exceptions.		
	Debugging	TRACE FLOW/VARS/OFF turn tracing on/off.		BREAK/UNBREAK line,line,... set or clear breakpoints.	
	Suspend execution	PAUSE (STOP acts like END)		BREAK (STOP acts like END)	
	Continue after STOP, break key or break point	CONT target		CON/CONTINUE line number	
		N/A	CONT or SST key		

Commands

Vendor		HP		TI	
	Model	HP-75	HP-71B with HP-IL	TI-74	CC-40
Display	Clear display	DISP CHR$(27)&"E"		PRINT or DISPLAY ERASE ALL	
	Output to display	DISP, PRINT	DISP, PRINT, implied DISP	DISPLAY (extended syntax), PRINT	
	Behavior of comma	Tabulate (21 chars), 5 items on display.		Tabulate (15 chars)	
	PRINT ends with ; or ,	Allowed		Allowed, disables clearing of rest of line.	
	Default display mode	Continuous with selectable DELAY		Continuous with selectable PAUSE	
	Continue after PRINT	ENTER key		CLR or ENTER key	
	Position cursor	PRINT/DISP TAB(n) Columns start at 1.		PRINT TAB(n) Columns start at 1.	
				DISPLAY AT(n) SIZE(s), TAB(x) TAB(x) is relative to AT(n).	
	Set display delay	DELAY seconds Accurate to 0.1s	DELAY line secs, scroll secs Sets both line and scroll delay. Values ≥ 8 are infinite, fractions of a second allowed.	PAUSE seconds or PAUSE ALL Accurate to 0.1s. Must be set in program. Inherited by procedure. Changes are local to procedure.	
	Display formatting	DISP USING"3A, 3D.DD";A$;X. Special HP-format instructions. Format can be put on IMAGE line and referenced by line number.		PRINT/DISPLAY USING "### ###.##";A$;X USING works for current statement only. Text uses same format characters as numbers. Constant text is allowed. Format can be put on IMAGE line and referenced by line number.	
		N/A	FIX d, SCI d, ENG d set default display precision for numbers.		
	Reverse (light on dark)	N/A		N/A	
	Graphics screen	N/A	132x8		
	Query dot or pattern		GDISP$ returns complete display. CHARSET$ returns defined characters.		
	Set/reset dot		N/A		
	Draw (filled) rectangle				
	Draw line or polygon				
	Graphical patterns		GDISP string set pattern in display. CHARSET string defines characters. String is taken as binary data.	Can be installed	CALL CHAR(c,"hex(16)") Defines char c ≤ 6. Patterns are horizontal.
	More graphics commands		N/A		CALL INDIC(indicator,state) sets the display indicators.
	(x,y) outside screen area				N/A
Printer	Printer interface and type	HP-IL printer. There are HP-IL interfaces to HP-IB or RS-232. Any printer supporting these interfaces can be connected.		PC-324 (matrix, id 12, DockBus) HX-1000 (pen plotter, id 10, HexBus) Printer 80 (matrix, id 16, HexBus) HX-3000 (serial/parallel, ids 20/50, HexBus) HexBus devices need adapter cable for TI-74. DockBus devices need adapter cable for CC-40.	
	Printer output	Redirected PLIST, LIST, PRINT or DISP		LIST"12" (12 is PC-324) OPEN#channel,"12",OUTPUT:PRINT#channel...	
	Redirect display to printer	PRINTER/DISPLAY IS 'device', '*' resets to display			
	Set width for printer output	PWIDTH n		Specify with OPEN: OPEN#channel,...,VARIABLE n	
	Set Printer to text or graphics mode	N/A		OPEN#channel,"10",OUTPUT:PRINT#channel,CHR$(x) x=17: text mode, x=19: graphics mode (HX-1000)	
	Printer commands in graphics mode			Use PRINT#channel,... to send plotter commands.	
	Additional printer commands in text mode			Settable options in OPEN after device number: OPEN#1,"10.S=0",OUTPUT sets small print.	

Commands

Vendor		HP		TI	
Model		HP-75	HP-71B with HP-IL	TI-74	CC-40
Sound	Beeper			N/A	DISPLAY BEEP ... ACCEPT BEEP ...
	Frequency range	Best accuracy is in the range 100..1400Hz. Default duration is 0.1s.			Only a single pitch available.
Input	Interactive data input	INPUT "prompt",default string;variable(s) Only one prompt and default string allowed.		INPUT prompt1;variable(s),prompt2;variable(s),... LINPUT prompt;var$ ACCEPT AT(n) SIZE(s) ERASE ALL VALIDATE("chars",keywords) NULL(def),var	
	Behavior of comma or semicolon after prompt	Comma separates default string from prompt. "?" appears if no prompt is given. Default string fills input buffer and can be edited by user.		Prompts may be expressions and must be followed by ;. Default is "? ". ERASE ALL deletes complete display instead of area set by AT and SIZE. Keywords are ALPHA, UALPHA, DIGIT, NUMERIC, ALPHANUM, ULPHANUM.	
	Allowed input values and keys	A comma seperated list of quoted or unquoted strings, numbers or expressions.		Numeric expression or (quoted) string. Unquoted strings are stripped.	
		END LINE enters one or more values (if separated by comma.) CONT leaves values unchanged.		ENTER or CTL+ENTER separates values. CTL+ENTER ignores input, ENTER alone sets NULL value (with ACCEPT).	
	Read keyboard directly	A$=KEY$, returns "" if no key pressed.	A$=KEY$, returns "" if no key pressed. KEYDOWN(key) checks if key is pressed. All keys checked if key isn't specified.	A$=KEY$, waits for a single key press. CALL KEY(code,status), status=0 if no key pressed, -1 if same as before, +1 if different key.	
	Some special key codes	unknown	f+Q: "fQ", g+Q: "q", END LINE: "#38", RUN: "#46", UP/DOWN: "#50","#51"	ENTER=13, LEFT/RIGHT=252/254, UP/DOWN=232/233, CLR=250 Key combinations with SHIFT, CTL, FN return combined codes (see manual.)	
	Read display contents as input	N/A	DISP$ returns display as string. Use VAL to parse the string.	N/A	
	DATA/READ/RESTORE	Data elements are quoted or unquoted strings or numeric constants. HP-71B: RESTORE target may be label / ON expr RESTORE targetlist.		Data elements are quoted or unquoted strings or numeric constants. RESTORE line numbers must be in the same program or subroutine.	
Program files	Tape filename syntax	No analog tape interface. The following commands hold for any file (RAM or external mass storage.) Filename syntax see page ProgVarsMem. Card reader has name "CARD", ":CARD" or "name:CARD". PCRD is an alias for CARD but creates a "private" file. The HP-75 allows suffix "/pass" to specify a password.		"1.name.NM" Name is 12 chars. Optional suffix .NM disables prompting messages. Name part can be omitted with OLD and defaults to first file found.	The Hex-Bus Wafertape behaves like the CI-7, except for the .NM suffix. The CI-7 is no "real" DockBus device and works only with the TI-74 or TI-95 calculators which contain the neccessary software logic.
	Other storage devices			"n.name or options", n is the device number: 8=QuickDisk, 20=RS-232, 100/101=PC interface. Filename syntax depends on device. PC-Interface uses the DOS convention 8+3 with complete path specification if neccessary ("\" is CTL+"/" = Yen).	
	Save program to tape in binary	COPY source TO destination Default source is current file.		SAVE "1.name.NM"	
	Save multiple programs	N/A		N/A	
	Set (password) protection	LOCK 'password' locks machine on power on. PRIVATE filename (or device :PCRD) makes file execute only (cannot be undone). PROTECT/UNPROTECT (un)protects a magnet card. HP-71B: SECURE/UNSECURE filename (re)sets write protection on a file.		SAVE "n.name",PROTECTED A protected file is execute only.	
	Save program to other device	COPY source TO destination Default source is current file.		SAVE "n.name" n=100 for PC interface.	
	Save in ASCII format	TRANSFORM source INTO TEXT destination HP-75 does not support destination name.		LIST "20.options" to serial interface. LIST "101.name" to PC interface.	
	Load binary program from tape	COPY source TO destination. Default destination is current file.		OLD "1.name.NM"	OLD "1.name"
	Load multiple programs	N/A		N/A	
	Load binary program from storage	COPY source TO destination. Default destination is current file.		OLD "n.name" n=100 for PC interface.	
	Load ASCII program	TRANSFORM source INTO BASIC destination HP-75 does not support destination name. Destination defaults to current file.		TI-BASIC cannot load an ASCII format BASIC program directly.	
	Load "foreign" program	Use LIF1 interchange format with TRANSFORM.		Use TIC74.EXE on PC with PC interface to create a binary file from source and load it with OLD"101.name".	

Commands

Vendor		HP		TI	
Model		HP-75	HP-71B with HP-IL	TI-74	CC-40
Program files	MERGE program lines	MERGE source,first line,last line Destination is the current file.		N/A	
	Handling of duplicate line numbers	Lines are replaced, file types must match.			
	Run program from storage or tape	CHAIN file RUN file,line number or label		RUN "n.name" Executable from program.	
	SAVE or LOAD special areas	Same internal files have special names: APPT stores the active appointments, KEYS stores the keyboard definitions. These are unquoted keywords.	Special file KEYS stores keyboard definitions. It is a standard filename and can be used with or without quotes.	Use LOAD.PGM delivered with PC interface to load machine language subroutines.	CALL LOAD("n.name") loads machine language programs.
	Check integrity of file	Automatically prompted for during write to card.		VERIFY "n.name"	
	Rename file	RENAME source TO destination		N/A	
	Delete file	PURGE file		DELETE "n.name" / CLOSE#channel,DELETE	
	Copy file	COPY source TO destination		USE OLD/SAVE	
	List directory	CAT device	CAT device / CAT$(number,device)	User program with CALL IO.	
	Format storage medium	INITIALIZE device,dir size	INITIALIZE volume device,dir size	FORMAT device (numeric id)	
Data files	OPEN channel on device or file	ASSIGN #channel TO 'name',type	CREATE type name,size,reclen ASSIGN #channel TO name	OPEN#channel,"n.name",access,type,VARIABLE n,mode #channel is #1..255, #0 is display or keyboard and always open. access is RELATIVE or defaults to sequential. type is DISPLAY or INTERNAL. n is the maximum record size. mode is one of INPUT/OUTPUT/APPEND/UPDATE. Defaults depend on the selected device. Options can appear in any order.	
	Valid OPEN modes and channels	File type is BASIC or TEXT, default is BASIC. #channel is #1..9999 BASIC files translate to a list of DATA statements with line numbers.	File type on CREATE is one of LIF1/TEXT/DATA/SDATA (see reference manual for details). #channel is #1..255 RESTORE #channel,large value sets pointer to eof for append.		
	Close channel	ASSIGN #channel TO * (or '*' or empty string) A file is closed if the channel is assigned to another file.		CLOSE #channel,DELETE DELETE purges closed file (device dependent.)	
	Write data sequentially	PRINT #channel;print items TEXT files and some devices use standard PRINT formatting.		PRINT #channel,print items DISPLAY files support formatting with comma, USING or TAB. INTERNAL files treat comma and semicolon the same and don't allow TAB. Features and exact format are device dependent. If the list ends with a delimiter, the output is pending.	
		No matrix support in output statements. HP-IL needs option ROM.	OUTPUT HP-IL device;print items Print items on both statements can be arrays A() or matrices M(,).		
	Read data sequentially	READ #channel;var1,var2,...		INPUT #channel,var1,var2,... LINPUT #channel,var$ Validity checking and features are device dependent. If the list ends with a delimiter, the input is pending. RESTORE #channel resets file pointer to first record. EOF(channel) tests for end of file.	
		No matrix support in input statements. HP-IL needs option ROM.	ENTER HP-IL device;var1,var2,... Variables in both statements can be arrays A() or matrices M(,).		
	Random access files	PRINT #channel,record;... READ #channel,record;... RESTORE #channel,record		OPEN#channel,"n.name",RELATVE,VARIABLE n,other options Each record is n bytes long. Record numbers range from 0 to 32767. PRINT #channel,REC rec,print items INPUT #channel,REC rec,var1,var2,... LINPUT #channel,REC rec,var$ RESTORE #channel,REC rec	
		Records are line numbers 1..9999. File type must be BASIC.	Record numbers are 0 to 1048575 Record size is defined by file.		
	Special I/O functions	Checking for end of file must be performed with either a user defined EOF record or with an ON ERROR handler. HP-IL knows many more I/O commands, some of which need a special ROM on the HP-75. The HP-IL commands in the HP-71B come with the interface.		EOF(channel) checks for end of file. CALL IO(device,status) performs control functions on HexBus/DockBus devices. "device" can be a number or a 12 byte string (control block.)	

Commands

	Vendor	Tandy Radio Shack	Canon	Epson
	Model	**TRS-80 Model 100**	**X-07**	**HX-20**
Program flow	**Labels**	N/A		
	Syntax for branch targets besides line numbers			
	ON ... GOTO/GOSUB	Available		
	IF ... THEN ...	THEN is mandatory except before GOTO.		
	IF ... THEN ... ELSE ...	Available		
	Nested IF	Allowed, nearest ELSE belongs to nested IF.		
	Multiline IF ... ENDIF	N/A		N/A
	WHILE ... WEND			Available
	REPEAT ... UNTIL			N/A
	SWITCH ... CASE ... ENDSWITCH			
	FOR I=1 TO 2 STEP -1 NEXT I	Loop executed once, I=0 after loop.		Loop not executed, I=1 after loop.
	Variable name on NEXT	Optional		
	Function and position of END	Executable command anywhere in program. Executable from keyboard.	Executable command anywhere in program. Closes all files. Executable from keyboard.	
Subroutines	**Local procedure definition besides GOSUB/RETURN**	N/A		
	Return from procedure			
	Variable scope			
	Call and parameter passing			
	Recursion			
Error handling	**ON ERROR**	ON ERROR GOTO		
	Error line and error code	ERL, ERR		
	Return from error handler	RESUME, RESUME NEXT, RESUME line number		
	Disable error handler	ON ERROR GOTO 0		
	More event handling	ON KEY GOSUB line number list ON COM/MDM GOSUB line number ON TIME$="time" GOSUB line number KEY/COM/MDM/TIME$ ON/OFF/STOP IPL "file.BA" defines startup program.	CONSOLE@1 enables alarm. ALM$="yyyy/mm/dd/day/HH/MM" sets alarm; each component can be omitted; day is a bit mask with 64=Sun..1=Sat. START$=+"startup sequence"	Monitor can set startup key sequence with K command.
	Debugging	N/A	TRON, TROFF	
	Suspend execution	STOP		
	Continue after STOP, break key or break point	CONT		

Commands

	Vendor	Tandy Radio Shack	Canon	Epson
	Model	**TRS-80 Model 100**	**X-07**	**HX-20**
Display	**Clear display**	CLS Text display scrolling clears graphics.	CLS Graphics is scrolled with text.	CLS (text only), GCLS (graphics only) Text display scrolling clears graphics.
	Output to display	PRINT, ?		
	Behavior of comma	Tabulate (15 chars)		
	PRINT ends with ; or ,	Allowed. ";" between items is optional.		
	Default display mode	Continuous output		
	Continue after PRINT			
	Position cursor	PRINT TAB(n), Columns start at 1.	PRINT TAB(n), Columns start at 0.	PRINT TAB(n);...;SPC(n);... Columns start at 1.
		PRINT @pos,... pos=40 is 2nd line, etc. POS/CSRLIN return x/y position.	LOCATE x,y positions cursor. POS/CSRLIN return x,y position.	LOCATE x,y,c (c=0,1 ?) LOCATES x,y positions virtual screen. POS/CSRLIN return x/y position.
	Set display delay	N/A	CONSOLE first line,size,f1,f2,f3 first line and size define scrollable area, f1=1 enables F-key display, f2, f3 control key click & repeat.	SCROLL speed,mode,xscrl,yscrl controls virtual screen scrolling. WIDTH cols,rows,scroll margin defines virtual screen size.
	Display formatting	PRINT USING"\ \ #,###.##";A$;X Works on current PRINT/LPRINT statement only. Mixed formats are allowed. "!" outputs single char, "&" formats a string with its exact length. "+" or "-" can be prefix or postfix, "$$", "**" and "**$" pad numbers to the left. Literal text can be escaped with "_".		
	Reverse (light on dark)	PRINT CHR$(27)"p";"Text";CHR$(27)"q"	N/A	
	Graphics screen	240x64	120*32	
	Query dot or pattern	N/A	POINT STEP(x,y) STEP makes coordinates relative.	POINT(x,y)
	Set/reset dot	PSET(x,y), PRESET(x,y)	PSET STEP(x,y), PRESET STEP(x,y) STEP makes coordinates relative.	PSET(x,y), PRESET(x,y)
	Draw (filled) rectangle	LINE(x,y)-(x,y),mode,BF F fills rectangle.	Use consecutive LINE commands.	
	Draw line or polygon	LINE(x,y)-(x,y),mode Continues if started with -. Bit 0 of mode=1 (set) or 0 (reset).	LINE STEP(..)-STEP(..) STEP makes coordinates relative.	LINE(x,y)-(x,y),mode Continues if started with -. mode=PSET (set) or PRESET (reset).
	Graphical patterns	N/A	FONT$(c)="c1,...,c8" defines character. c=128..159,224..255; c1..c8 may be constants or variables. CONSOLE@,1 resets all chars to default.	N/A
	More graphics commands	SCREEN 0,1 protects the last (label) line against scrolling.	CIRCLE STEP(x,y),r STEP makes coordinates relative.	COLOR fg,bg,set sets color on external screen. SCREEN text,graph controls external display. 0,0 is default (LCD).
	(x,y) outside screen area	FC Error	Clipping	Virtual screen with clipping
Printer	**Printer interface and type**	Centronics (any type).	Centronics with legacy plug. Plotter X-710 supported with special LPRINT syntax. Serial (TTL): X-711 thermal printer.	Built in dot matrix with graphics.
	Printer output	LPRINT, LLIST, PRINT#1,... after OPEN"LPT:" FOR OUTPUT AS #1	LPRINT, LLIST, PRINT#1,... after INIT#1,"LPT:" (or "GPR:" or "PRT:")	LPRINT, LLIST, PRINT#1,... after OPEN"O",#1,"LPT0:"
	Redirect display to printer	OPEN either "LPT:" or "LCD:"	INIT one of "LPT:", "GPR:", "PRT:" or "CON:"	OPEN either "LPT0:" or "SCRN:"
	Set width for printer output	N/A; LPOS returns current position.	N/A	WIDTH "device", n
	Set Printer to text or graphics mode	Depends on printer.	LPRINT CHR$(18); sets X-711 to graphics mode; LPRINT CHR$(13);CHR$(17); sets text mode.	N/A
	Printer commands in graphics mode	Send commands with LPRINT.		COPY prints text and graphic screen on built in printer. Only the visible area is printed.
	Additional printer commands in text mode	LCOPY copies the text screen. SAVE"LPT:" is the same as LLIST. LPOS returns current column.	LPRINT[size,color] ... (see PRINT) size: 1..16 color: 0..3	SAVE"LPT0:" is the same as LLIST.

Commands

Vendor		Tandy Radio Shack	Canon	Epson
Model		TRS-80 Model 100	X-07	HX-20
Sound	Beeper	BEEP, SOUND pitch,duration Duration of 50 is 1 second.	BEEP pitch,duration Duration of 20 is 1 second.	SOUND pitch,duration Duration of 10 is 1 second.
	Frequency range	0..16383 (useful: 220..16383) Frequency=4915680 Hz / pitch 5586: 880 Hz	0: pause 1..48: halftones starting from "do" 49..4095: frequency=19200 Hz / pitch.	1..28: tones C to B in 4 octaves 29..59: halftones, 0: pause 13: 880 Hz
Input	Interactive data input	INPUT "prompt";variable(s) Only one prompt allowed.		
	Behavior of comma or semicolon after prompt	Comma is not allowed. "? " is always added to the prompt.		Comma suppresses "?" after prompt.
	Allowed input values and keys	Number or (quoted) string. Unquoted strings are stripped on left end.		Number or (quoted) string. Unquoted strings are stripped on both ends.
		Values are separated by commas or ENTER. Empty input leaves values untouched.		Values are separated by commas. Empty input causes ?Redo message except for a single string variable.
	Read keyboard directly	A$=INKEY$, returns "" if no key pressed.		
		A$=INPUT$(count) returns exactly count key presses.	INIT#1,"KBD:" opens keyboard as file. A=INP(#1) waits & returns ASCII code. A=SNS(#1) returns 0 or ASCII code. A=STICK(0) returns status of cursor keys. A=STRIG(0) returns status of space bar A=STRIG(1) returns status of F6.	A$=INPUT$(count) returns exactly count key presses.
	Some special key codes	ENTER=13, LEFT/RIGHT=29/28, UP/DOWN=30/31, BS=8 Use ON KEY to read function keys.	ENTER=13, LEFT/RIGHT=29/28, UP/DOWN=30/31, HOME/CLR=11/12 F-Keys return strings. STICK(0) gives a value from 0 to 8 (up=1, up+right=2,...).	ENTER=13, LEFT/RIGHT=29/28, UP/DOWN=30/31, BS=8, CLR=12 PF-Keys return strings.
	Read display contents as input	N/A	A=SCREEN(x,y) returns ASCII code.	N/A
	DATA/READ/RESTORE	Data elements are quoted or unquoted strings or numeric constants.		
Program files	Tape filename syntax	"name" (6 chars) with CSAVE, CLOAD, etc. "CAS:name" with SAVE, LOAD, MERGE, OPEN. Name can be omitted on load and defaults to first file found.	"name" (6 chars) with CSAVE, CLOAD, etc. "CASO:name" (output) or "CASI:name" (input) with SAVE, LOAD, INIT. Name can be omitted on load and defaults to first file found.	"CAS0:name" (8 chars for name). "CAS1:name" for external tape. Prefix CAS0: can be omitted. Name can be omitted on load and defaults to first file found.
	Other storage devices	"RAM:name6.ex" is a RAM disk file. The prefix can be omitted. .ex is .DO for text, .BA for BASIC, .CO for ml. Serial I/O: "COM:RBPSX" Floppy: "0:name6.ex" or "1:..."	"RAM:name6","t" is a RAM disk file of type "t". Type "P" is a BASIC file. Serial I/O: "COM:" (TTL) or "OPT:" (optical coupler)	ROM cartridge: "PAC0:name" Serial I/O: "COM:(RBPSF)" Floppy: "d:name", d = A/B/C/D File name convention for floppy is unknown.
	Save program to tape in binary	CSAVE "name" SAVE "CAS:name"	CSAVE "name" SAVE "CASO:name"	SAVE"name"
	Save multiple programs	N/A		
	Set (password) protection	N/A		TITLE "name" makes area read only.
	Save program to other device	SAVE "device:name"	SAVE "device:name",baud,"mode" SAVE "device:name",size,"type"	SAVE "device:name"
	Save in ASCII format	SAVE "device:name",A	INIT#1,"Device:name",p1,"p2" LIST#1	
	Load binary program from tape	CLOAD "name" LOAD "CAS:name"	CLOAD "name" LOAD "CASI:name"	LOAD "name"
	Load multiple programs	N/A		
	Load binary program from storage	LOAD "name",R R starts program. Format is detected.	LOAD"name" (RAM disk) LOAD"device:name",p1,"p2" (device)	LOAD "name",R R starts program. Format is detected.
	Load ASCII program		INIT#5,"device:name",p1,"p2" CALL &HEE1F turns on remote control CALL &HEE33 turns off remote control	
	Load "foreign" program	Use ASCII mode serial I/O.		

Commands

	Vendor	Tandy Radio Shack	Canon	Epson
	Model	**TRS-80 Model 100**	**X-07**	**HX-20**
Program files	**MERGE program lines**	MERGE "name" ASCII only	Same as Load ASCII via remote control.	MERGE "name" ASCII only
	Handling of duplicate line numbers	Lines are replaced.		
	Run program from storage or tape	LOAD "name",R	RUN "name" works only for files in RAM.	LOAD "name",R
	SAVE or LOAD special areas	CSAVEM "name",start,end,entry SAVEM "name",start,end,entry	N/A	SAVEM"name",start,end,entry
		Text editor and telco do load/save		
	Check integrity of file	CLOAD? "name"		SAVE"name",V verifies after save
		LOAD? "name"		LOAD? "name"
	Rename file	NAME source AS destination	N/A	NAME source AS destination
	Delete file	KILL "file"	DELETE "file","type"	KILL "file"
	Copy file	Use LOAD/SAVE		
	List directory	FILES "device:"	DIR (RAM only)	FILES "device:"
	Format storage medium	Use DOS menu	N/A	FORMT "d:"
Data files	**OPEN channel on device or file**	OPEN "name" FOR mode AS #channel	INIT#channel,"name",param1,"param2" param1 is baudrate or size param2 is mode or file type.	OPEN"mode",#channel,"name" DEFFIL reclen,address defines RAM file. Address must be in area defined by CLEAR.
	Valid OPEN modes and channels	INPUT/OUTPUT/APPEND, #1..255 Total number of files set with MAXFILES=n	Mode depends on device. RAM Files are always I/O. #channel is #1..5. #5 can be used for remote control.	I/O/R, #1..16, mode R on disk only. FILNUM n reserves space for up to 15 floppy disk FCBs. RAM files with DEFFIL are unnamed, have random access and must be managed by program.
	Close channel	CLOSE closes all channels CLOSE #ch1,#ch2,... closes selected channels.	There is no close, just reopen the channel.	CLOSE closes all channels CLOSE #ch1,#ch2,... closes selected channels.
	Write data sequentially	PRINT #channel,print items Formatting with comma, USING, SPC or TAB is possible.		
		N/A	OUT #channel,code outputs a single byte	N/A
	Read data sequentially	INPUT #channel, var1, var2, ... LINE INPUT #channel, var$		
		var$=INPUT$(count,#channel)	A=INP(#channel) reads single byte, waits for input. A=SNS(#channel) reads single byte without waiting. 0 denotes no input.	var$=INPUT$(count,#channel)
	Random access files	N/A		FIELD #channel,len AS var$,... (Record size is 128 bytes) LSET/RSET var$=string CVI/CVD/CVS/MKI$/MKD$/MKS$ PUT/GET #channel,record
				PUT%/GET% record,var1,var2,...,var$ Write/read RAM file. Only one string allowed at end of list.
	Special I/O functions	EOF(channel) tests for end of file.	N/A	LOF(channel) returns length of file or chars left in input buffer. LOC(channel) returns current record (mode R) or sector (I/O).

Special commands and some remarks

Casio FX-730/770/780/785/795P MODE commands

See next page.

Casio VX-4

PEEK/POKE only work in RAM: &H1000..02FFF in segment 0 on machines without RAM extension.
MODE110 = CALL

Casio BASIC ROM OM-53B for PC-2000C/AI-1000

The following information was posted in the french MySilicium forum:

The differences between PB-1000 BASIC and OM-53B BASIC for the PB-2000C/AI-1000:

CALL, TIME$ and DATE$ were removed. 6 commands were added, but 5 of them are hidden behind the SYSTEM command.

- RENUM: Well known.

- SYSTEM CALL: Identical to CALL on PB-1000.

- SYSTEM SET: Writes a sequence of key codes (00-99) into the key buffer.

- SYSTEM SW: Calls the RS232C settings menu similar to the VX-4 F.COM>Device>Switch menu.

- SYSTEM COPY: Copies a complete MD-100 disk to another disk.

- SYSTEM SUM: ROM card sum and xor test.

Sharp display routines

The following information was posted in the french MySilicium forum:

The display of the following Sharp PCs goes blank during a calculation:
1210, 1211, 1212, 1245, 1246, 1247, 1248, 1250, 1251, 1260, 1261, 1262, 1401, 1402, 1403

Workaround:

- 1245, 1250, 1251: CALL &11E0 (with WAIT 0) turns display on (with some stray pixels), CALL &11E5 turns it off.

- 1260, 1261, 1262: Display contents stays visible if followed by a ";".

- 1401, 1402: CALL &5A2 turns display on, CALL &59E turns it off. (cf. Le Sharpentier #9, page 30.)

- 1403: CALL &4B8 turns LCD on, CALL &4B4 turns it off.

- 1210, 1211, 1212: Impossible to turn on display. A hidden command # allows some graphics in the first 3 columns.
(See l'Ordinateur de Poche #9, page 63 and #13, page 50 for an example).

Sharp PC-1260/61/62 graphics

The following information is from the book: "System und Trickbuch für den SHARP PC 1260/61" by Bernd Saretz

The display is divided in four areas of 12x5x7 dots each:

Upper left: &2000..&203B (8192..8251)
Upper right: &2800..&283B (10240..10299)
Lower left: &2040..&207B (8256..8315)
Lower right: &2840..&287B (10304..10363)

The dots are set with POKE. Bit 0 is top, bit 6 is bottom.

Address &203D (8253) controls the indicators. Bit 3 selects Kanji mode and locks you out!

The display must be turned on by CALL &A907 (stays on after break!) or a small ML routine:

10 POKE 25000,2,1,229,164,55:CALL 25000:WAIT 0
On a 1262 the address 25000 might not be the best idea.

Sharp PC-E220/G8xx

CALL has an additional parameter to call Software in other ROM banks which start at &HC000

CALL #5,&HC000 starts the built in Z80-Assembler, even on the G820 where this is officially not supported.

Casio FX-730/770/780/785/795P MODE commands

The following information was posted in the french MySilicium forum:

On some CASIO pockets an equivalent to PEEK and POKE exists.

MODE19(A,B) works like POKE A,B
MODE18(A,B$) is similar to PEEK A

An example how to use MODE18:

```
10 A=1234
20 MODE18(A,B$)
30 B$="&H"+B$
40 PRINT VAL(B$)
```

The FX-795 has more hidden functions use by the library program. Most of them will not work on other machines.

Keyboard scan

MODE 21,A,B waits for key and returns internal code in A$ and B.

Matrix operations

MODE 92,A,B : copy matrix A to matrix B.
MODE 93,A : transpose matrix A.
MODE 94,A,B,C : matrix multiplication C=A*B.
MODE 97,A,X,Y : get dimensions X, Y of matrix A.

MODE 99 controls the ERROR stop of a program.

```
10 MODE 99,1
20 PRINT 1/0:REM error is ignored
30 MODE 99,0
40 PRINT 1/0:REM creates error
```

A lengthy list of MODE commands has been posted by member *ynopum* here:

After examining of the quoted program library, and also reading of the forum, and playing with my Casio, I tried to make a list of the undocumented MODE commands. Most of them were already known, but some are not. My tests showed that the matrices sizes are not limited to 9x9 as in the FXLibrary. Successful operations were done with 15x15 size for matrix inversion. We should take in mind that for not well defined matrices the inversion can give unreliable results. Also I did some speed tests for the matrix operations. The "internal" matrix commands work roughly 5~6 times faster than BASIC written routines doing the same job (of course a difference of the algorithm plays a major role maybe). This was tested with random filled matrices 5x5. Also I tested the calculator fx-5500LA with such 5x5 random matrices. It seems it is about twice faster than fx-795P. It is pity it is not programmable, and the fx-4500PA is obviously much slower than 5500LA - I decided not to make tests on it. So, here is the list for now:

MODE 10
Standard truncation of the last digits after calculation. This mode is reset after turn-off/on of the device. Produces round results for integers. (thanks to Xerxes)

MODE 11
No truncation of the last digits. This mode is canceled after turn-off/on of the device. Should be used if we suspect error accumulation in default mode. (more explanation on Page11 of the thread)

MODE18(A,B)
Gets from address A the HEX value as B$. The syntax is with brackets!

MODE19(A,B)
Puts in address A the value B. The syntax is with brackets!

MODE 20,A
Evaluates as expression the variable $ and puts the result in variable A

MODE 21,A,B
Waits for keypress and returns its character in A$ and its DEC code in B

MODE 22 unknown
MODE 23 unknown
MODE 24 unknown
MODE 25 unknown

MODE 26,A
Converts the value of $ variable (assumed to be a BIN number) to DEC variable A

MODE 27,A
Converts the value of $ variable (assumed to be a HEX number) to DEC variable A

MODE 28,A
Converts the value of variable A (DEC number) to LONG BIN string in variable $

MODE 29,A
Converts the value of variable A (DEC number) to SHORT BIN string in variable $

MODE 90,A,B,F
Matrix operation: inverse of matrix A goes to matrix B. Return code F is for success. If the value of F = 0 then the inversion operation was unsuccessful.

MODE 91,A,,D
Matrix operation: determinant of matrix A goes to variable D. Note the double comma! I tried to put variable there but an Error occurs.

MODE 92,A,B
Matrix operation: contents of matrix A goes to matrix B

MODE 93,A
Matrix operation: contents of matrix A is transposed

MODE 94,A,B,C
Matrix operation: matrix A multiplied by matrix B goes to C. C variable should not be an array. The command will create it as an array.

MODE 95,A
Equal to INPUT,A (used in the FXLibrary with error code reading from the memory)

MODE 96,Oper,A,B,C
Boolean operation. Oper has the following options
0 is Twos complement,
1 is NOR,
2 is AND,
3 is OR,
4 is XOR

MODE 97,A,X,Y
Matrix operation: dimensions of matrix A go to X an Y

MODE 99,0 Breaks on Error (default behavior)
MODE 99,1 Continues execution on Error

I tried to find the address of the special $ variable. It seems to start from address 400.
The last entry line is at address 528. The FXLibrary is at address about 16400.
I also tried the "password erase" offered by Xerxes - it works. The password string is located at address 308.
The variables A-Z are stored backwards from the end of the memory: variable A is at address 16376. Variable B is 8 bits before at 16368 etc. Variable Z is at 16176.
The MEMO-databank seems to be at address 588.
The IN-OUT-CALC string seems to be after the program area - not at fixed address.

Well, another curious thing is that I managed to use 6 more characters from the code table (small D, small L, -1 index, thick /, b/, /c). I did this with using MODE19 to the address of a string variable. Maybe one day somebody will need to use those special chars. Functions like CHR$ and ASC can be substituted by short (1-row) programs, using MODE18 & MODE19.

Also the scan-codes of the special buttons for MODE21 are:

128 - SIN
129 - COS
130 - TAN
134 - LOG
135 - LN
136 - EXP
137 - SQR (square root sign)
152 - DEG(
182 - &H
183 - CUR (cubic root sign)
185 - HYP
205 - X^2
206 - X^3
207 - 10^x
219 - CLS
220 - ENG
222 - STAT
234 - MEMO
235 - EXT
239 - EXE
240 - INS
241 ->
242 <-
244 - STOP
245 - MODE
246 - ^R (return of the last entered line)
247 - Shift
251 - IN
252 - OUT
253 - CALC

The rest of the codes correspond to the letter or sign of the button depending on the char-code table. I.e. for the key "G" we have 38, and of we are in EXT-mode the code will be 70 (for "g"). For the key "+" the code is 1. Button BRK doesn't give code as the program execution breaks when it is pressed.