

# JPC ROM

## Quick Reference Guide

### **ADBUF\$ ( *buffer id* )**

Function : Returns the address of the buffer specified by its identification number. The following table lists various buffers used by the system :

- 808 : Hold a string of characters used by STARTUP,
- 83D : MARGIN setting,
- 83E : Hold a string of characters used by ENDUP,
- BFB : Character set defined by CHARSET, and
- BFC : Address of Lex files.

### **ADCREATE *file* [ , *password* ]**

Statement : Create an empty address file. A card is composed of the following fields :

- name and first name, separated by a /,
- phone number,
- 4 lines to store the address,
- a line to store general informations, and
- a line to store a criterion to be used by your own programs.

### **ADDELETE *file* , *number* [ , *password* ]**

Statement : Removes a card from an address file.

### **ADFIND ( *file* , *string* [ , *password* ] )**

Function : Looks for a name in an address file and returns the number of the card. Rules used during search are :

- name only (without /),
- name and first name (with /), and
- abbreviated search (name terminated with a dot).

### **ADGET *file* , *array* , *number* [ , *password* ]**

Statement : Reads a card and stores it into a string array.

**PPC Paris**

B.P. 604 - 75028 Paris Cedex 01 - France

(c) PPC Paris, 1988

**ADPUT** *file* , *array* [ , *password* ]

Statement : Write a card (a string array) into an address file.

**ADSIZE** ( *file* [ , *password* ] )

Function : Returns the number of cards in an address file.

**ARR** ( *n* , *p* ) *PERM* (  $\wedge$  *p* )

Function : Compute the number of possible different arrangements (permutations) of *n* items taken *p* at a time.

**ASC\$** ( *string* )

Function : Returns a string stripped of all non-displayable ASCII characters.

**ATH\$** ( *string* [ , *mode* ] )

Function : Returns the hexadecimal string corresponding to the parameter string. If *mode* = 1, nibbles in a byte are not reversed.

**ATTN ON / OFF**

Statement : Enables or disables the action of the [ATTN] key to stop program execution.

**BELL**

Statement : Causes the printer's beeper to sound if possible.

**BOLD ON / OFF**

Statement : Enables or disables the bold mode of the printer.

**CASE** *element* , ...

**CASE** *relational operator element* , ...

**CASE** *element TO element* , ...

**CASE ELSE**

Statement : Part of SELECT ... CASE ... END SELECT structure.

**CENTERS** ( *string* , *width* )

Function : Adds spaces at the beginning of the string specified in parameter in order to center it.

**CESURE** ( *string* , *width* )

Function : Returns the position of the first place in the string where a word-break can occur.

**COMB** ( *n* , *p* )

Function : Computes the number of possible different sets of *n* items taken *p* at a time.

**CONTRAST**

Function : Returns the current contrast setting.

**DATEADD** ( *date* , *days* )

Function : Computes the date corresponding to the specified date incremented by the specified number of days.

**DATESTR\$** ( *date* )

Function : Converts a date to the HP-71 string format for date : "yyyy/mm/dd".

**DBLIST** [ *file* [ , *start line* [ , *final line* ] ] ]  
[ *INDENT indentation* ] [ **TO target** ]

Statement : Produces a structured listing of a Basic program.

**DDAYS** ( *date1* , *date2* )

Function : Compute the number of days between two dates.

**DDIR** [ *file specifier* ] [ **TO target** ]

**DDIR ALL** [ **TO target** ]

Statement : Lists directory of the specified device.

**DMY**

Statement : Enable date input in numeric format *dd.mm.yyyy*.

**DOW** [ ( *date* ) ]

Function : Returns the day of week corresponding to the specified date parameter.

**DOWS** [ ( *date* ) ]

Function : Returns the name of the day corresponding to the specified date or today.

**EDIT** [ *file1* ] [ **TO** *file2* ]

Statement : Allows merging of Lex files, or editing files on external peripherals. Nonprogrammable.

**ENDUP** *command string*

Statement : Defines a command string to be executed when the HP-71 turns off.

**ENDUP\$**

Function : Returns the command string specified in ENDUP.

**ENTRY\$** ( *keyword* [ , *sequence* ] )

Function : Returns the entry point address for the specified keyword.

**ESC\$** [ ( *string* ) ]

Function : Returns the string with a leading "escape" character.

**EXECUTE** *command string*

Statement : Executes the specified command string and stops program execution.

**EXIT** *loop variable*

Statement : Exit a FOR ... NEXT loop.

**FILESIZE** ( *file* )

Function : Returns the size in bytes of the specified file.

**FIND** *string*

Statement : Finds a character string in a Basic program. Nonprogrammable.

**FINPUT** *input* , *prompt* [ , *format* ] , *attn*

Statement : Creates an input mask and waits for data input from the user.

**FKEY** *key*

Statement : Insert a key code at the beginning of the keyboard buffer.

**FORMAT\$** ( *string* , *width* )

Function : Inserts extra spaces inside a string so that it will have exactly the specified number of characters.

**FPRIM** ( *argument* [ , *direction* ] )

Function : Returns the first prime number after the argument.

**FRAC\$** ( *real number* [ , *accuracy* ] )

Function : Approximates a real number by a fraction.

**GLINE** *x* , *length* , *first* , *size* , *gap*

Statement : Builds a raster graphics representation of a drawn line for use with ThinkJet or LaserJet printers.

**GPSET** *x*

Statement : Prepares drawing of a pixel on ThinkJet or LaserJet printers.

**HMS** ( *argument* )

Function : Converts decimal hour or degree data into an equivalent value in HMS format.

**HMSADD** ( *arg1* , *arg2* )

Function : Returns the sum of two arguments interpreted using the HMS format.

## **HMSSUB ( *arg1* , *arg2* )**

Function : Returns the difference of two arguments interpreted using the HMS format.

## **HR ( *argument* )**

Function : Converts a number from HMS format to its decimal equivalent.

## **HTAS ( *hexadecimal string* [ , *mode* ] )**

Function : Converts a string of hexadecimal digits into an ASCII character string. If *mode* = 1, nibbles in a byte are not reversed.

## **IF *logical expression* THEN**

*program segment*

## **END IF**

or :

## **IF *logical expression* THEN**

*program segment*

## **ELSE**

*program segment*

## **END IF**

Statement : Extends the standard IF structure to allow multiple line statements.

## **INVERSE [ *begin* , *end* ]**

Statement : Displays the binary complement of the contents of the LCD.

## **KA [ *file* ]**

Statement : Interactive address directory editor.

Keystrokes defined are :

- [ATTN] : exit KA,
- [(], [)], [g]([( and [g][]) ] : move inside the file,
- [v], [^], [g][v] and [g][^] : move inside the card,
- [<], [>], [g][<] and [g][>] : scroll the display,
- [0] to [7] : direct access to a card field,
- [f][CAT] : display the number of cards,
- [f][DELETE] : delete the current card,
- [f][EDIT] : edit the current card,
- [f][INPUT] : input a new card,
- [f][KEY] : input a password, and
- [A] to [Z] : looks for a name.

## **KEYWAIT\$**

Function : Waits until a key is pressed and then returns a string representing its keycode.

## **LEAVE**

Statement : Exits from a structured programming loop such as **WHILE**, **REPEAT** or **LOOP**.

## **LEX *file* ON / OFF**

Statement : Enables or disables a Lex file.

## **LOOP**

*program segment*

## **END LOOP**

Statement : Defines an endless loop.

## **MAP *file* , *string1* , *string2* [ , *from* [ , *to* ] ]**

## **MAP # *channel* , *string1* , *string2* [ , *from* [ , *to* ] ]**

Statement : Applies a mapping function to the contents of a text file.

## **MAP\$ ( *string1* , *string2* , *string3* )**

Function : Applies a mapping function to the contents of a character string.

**MARGIN** [ *position* ]

Statement : Enables a beep when the cursor reaches the specified position, or disables it when *position* is missing or 0.

**MAXD** ( *device specifier* )

Function : Returns the maximum number of entries that can be stored in the directory of a mass storage medium.

**MAXM** ( *device specifier* )

Function : Returns the maximum storage capacity available on the medium.

**MDY**

Statement : Enables date input in numeric format *mm.dyyyy*.

**MEMD** ( *device specifier* )

Function : Returns the number of entries in the directory of the specified medium that remain available for new files.

**MEMM** ( *device specifier* )

Function : Returns the available room in the file storage area of the specified medium.

**MENU** ( *number of elements* [ , *first element* ] )

Function : Read DATA and display them to create interactive menu facility. Following keystrokes are defined :

- [ATTN] : exit MENU,
- [V], [^], [g][v] and [g][^] : move inside the menu,
- [ENDLINE] : validates the displayed item.

**MERGE** *file* [ , *first line* [ , *last line* ] ]

Statement : Extends the standard keyword to Lex files. Nonprogrammable.

**MODE** *argument*

Statement : Changes the print pitch on the printer.

**NEXTOP\$** ( *hexadecimal address pointer* )

Function : Returns the address of the next assembler instruction.

**NLOOP** [ ( *loop number* ) ]

Function : Returns the number of devices on the HP-IL loop.

**NPRIM** ( *n1* , *n2* )

Function : Returns the number of prime numbers in an interval.

**OPCODE\$** ( *hexadecimal address* )

Function : Returns the mnemonic of the machine language instruction pointed to by the specified address.

**PAGELEN** [ *page length* [ , *text length* ] ]

Statement : Sets the page and text lengths on the printer.

**PAINT** ( [ *state* , ] *x* , *y* )

Function : Turns on a pixel on the HP-71 display and returns its value before modification.

**PARPOLL** [ ( *loop number* ) ]

Function : Returns the result of an HP-IL loop parallel poll.

**PBLIST** [ *file* [ , *start line* [ , *final line* ] ] ]

[ *INDENT indentation* ] [ **TO** *target* ]

Statement : Produces a structured listing of a Basic program on the current printer device.

**PCR**

Statement : Moves the print head to the beginning of the line.

**PDIR** [ *file specifier* ] [ **TO** *target* ]

**PDIR ALL** [ **TO** *target* ]

Statement : Prints directory of the specified device.

**PEEK\$** ( *hexadecimal address* , *number of nibbles* )

Function : Returns the contents of a memory area specified by its address.

**PERF ON / OFF**

Statement : Enables or disables the *perforation skip* mode on the current printer device.

**PFF**

Statement : Advances paper to the beginning of next page.

**PGCD** ( *arg<sub>1</sub>* , *arg<sub>2</sub>* [ , *arg<sub>3</sub>* [ , ... *arg<sub>10</sub>* ]... ] )

Function : Computes the greatest common divisor of two or more numbers.

**PHI** ( *argument* )

Function : Returns the number of integers between 1 and *argument* that are relatively prime to *argument*.

**PLF** [ *number of lines* ]

Statement : Advances the paper by the number of lines specified.

**POKE** *hexadecimal address* , *data*

Statement : Writes to memory at the specified hexadecimal address.

**POS** ( *string* , *min* [ , *max* ] )

Function : Returns the position in a string of the first character whose value falls within a specified range. *Min* and *max* can be specified either as a decimal number or as a character.

**PPCM** ( *arg<sub>1</sub>* , *arg<sub>2</sub>* [ , *arg<sub>3</sub>* [ , ... *arg<sub>10</sub>* ]... ] )

Function : Returns the smallest common multiple of all arguments.

**PRIM** ( *number* )**PRIM** ( *higher part* , *lower part* )

Function : Returns 0 if a number is prime, or the smallest divisor of that number.

**REDS** ( *string* )

Function : Trims all leading and trailing spaces from the specified string.

**REDUCES** ( *string* )

Function : Reduces all substrings consisting of two or more spaces to a single space, and removes leading and trailing spaces.

**RENUMREM** [ *new start* [ , *increment* [ ,  
*old start* [ , *old end* ]]] ]

Statement : Renumbers a Basic program with special handling for comment lines.

**REPEAT**

*program segment*

**UNTIL** *logical expression*

Statement : Defines a loop which is repeated until the logical expression evaluated by UNTIL statement is true.

**REPLACES** ( *string* , *pattern1* , *pattern2* [ , *start* ] )**REPLACES** ( *string* , *pattern1* , *pattern2* , *wild* )

Function : Replaces a substring with another in the target string using HP text editor rules (first syntax) or a wild card character (second syntax). Text editor rules are :

- . : any character,
- @ : any number of unspecified characters,
- & : the text that matches pattern1 when used in pattern2,
- ^ : beginning of a line (must be the first character in pattern1),
- \$ : end of a line (must be the last character in pattern1),
- and
- \ : cancel the meaning of the previous \.

## ROMAN ON / OFF

Statement : Enables the *Roman* extended character set (see table below).

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
A																
B																
C																
D																
E																
F																

## RREC\$ ( address , device specifier )

Function : Reads a record from the specified mass storage device.

## SELECT *expression*

**CASE** *match item*

*program segment*

**CASE** *match item*

*program segment*

:

**CASE ELSE**

*program segment* ]

## END SELECT

Statement : Provides conditional execution of program segments. See **CASE** for *match item* syntax.

## SHRINK *file*

Statement : Minimizes the size of a text file in Ram, releasing memory that is not used to store text.

## SLEEP

Statement : Puts the HP-71 into light sleep mode.

## SPACES\$ ( [ character / string , ] repeat )

Function : Returns a string consisting of the specified number of characters of strings (or spaces, default value).

## SRQ [ ( loop number ) ]

Function : Sends a identification message on the HP-IL loop to check whether a peripheral requires service.

## STACK *number of levels*

Statement : Sets the size of the command stack to the specified number of levels.

## STARTUP\$

Function : Returns the STARTUP command string.

### **SYSEDIT** *hexadecimal address*

Statement : Puts the HP-71 into an interactive memory editor / disassembler mode. Following keystrokes are defined :

- [ATTN] or [F][OFF] : Exit SYSEDIT,
- [+], [,], [\*] or [/] : Move the editor window through memory,
- [A][1] to [A][8] : NIBASC,
- [N][1] to [N][9] et [N][.][0] à [N][.][6] : NIBHEX,
- [C][1] to [C][6] : Decimal constant,
- [C][H][1] to [C][H][6] : Hexadecimal constant,
- [R][1] to [R][5] : Relative address,
- [H] : Hexadecimal mode,
- [D] : Disassembler mode,
- [L] : LCASC if disassembler mode active,
- [F] : Saving disassembler output,
- [=] : Direct move,
- [I] : Move and push address,
- [J] : Return,
- [ENDLINE] : Validation,
- [Z] : Address editing, and
- [F][Z] or [M] : Memory editing.

### **TOKEN** ( *keyword* [ , *sequence* ] )

Function : Returns the Lex Id and token for the specified keyword.

### **UNDERLINE ON / OFF**

Statement : Enables or disables underline mode on the printer.

### **VARSWAP** *variable1* , *variable2*

Statement : Swaps the contents of two variables or array elements.

### **WHILE** *logical expression*

*program segment*

### **END WHILE**

Statement : Defines a loop which is executed as long as *logical expression* is true.

### **WRAP ON / OFF**

Statement : Enables or disable the printer wrap-around mode.

### **WREC** *sector* , *address* , *device specifier*

Statement : Writes a 256 bytes string to the specified sector of selected mass memory device.