

System 35/45 Programming Language

HP Enhanced BASIC Statements and Functions

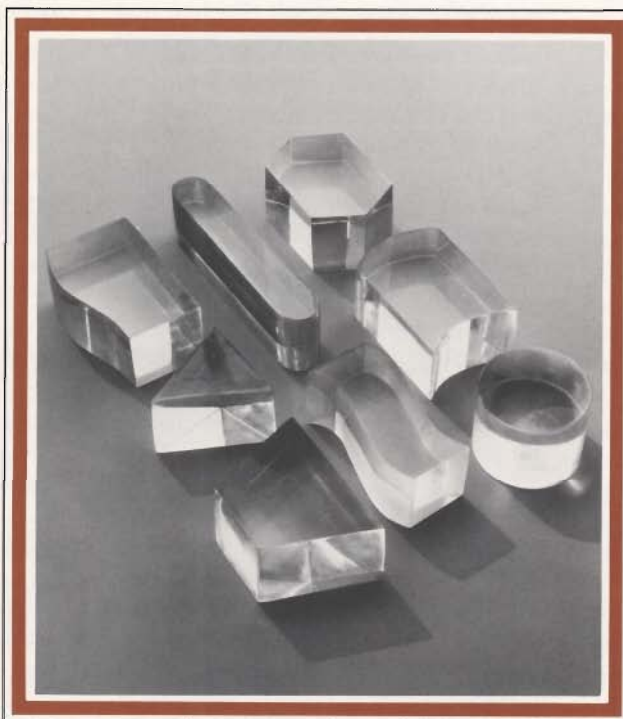
This technical supplement includes a complete list of the System 35's and System 45's HP enhanced BASIC operators, functions and statements. Optional statements that become available with the addition of the appropriate system ROM (Graphics, I/O, Advanced Programming, etc.) are also defined in this document.

The System 35 and 45 utilize an enhanced version of the BASIC programming language. Extending beyond the proposed ANSI standard minimal BASIC, HP enhanced BASIC includes array operations, debugging tools, subprograms, multicharacter variables and line identifiers, formatted outputs and annotated documentation in programs.

System 35/45 Mainframe Statements and Functions

Operators

Numerical	Relational	Logical	String
+	=	AND	&
-	>	OR	
.	<	NOT	
/	>=	EXOR	
DIV	<=		
MOD	<>		
^			
=			



Functions

System Functions

ABS — returns the absolute value of the numeric expression.

ACS — returns the principal value of the arccosine of the numeric expression expressed in the current angular unit mode.

ASN — returns the principal value of the arcsine of the numeric expression expressed in the current angular unit mode.

ATN — returns the principal value of the arctangent of the numeric expression expressed in the current angular unit mode.

COS — returns the cosine of the angle which is represented by the numeric expression.

DROUND — returns the numeric expression rounded to the specified number of significant digits.

ERRL — returns the line number of the statement which had the error.

ERRN — returns the error number.

EXP — returns the value of Napierian e raised to the power of the computed expression.

FRACT — returns the fractional part of the evaluated expression and is defined by the formula.

INT — returns the greatest integer which is less than or equal to the evaluated expression.

LGT — returns the common logarithm (base 10) of a positive numeric expression.

LOG — returns the natural logarithm (base e) of a positive numeric expression.

MAX — returns the greatest value in the list.

MIN — returns the smallest value in the list.

PI — returns the value of π .

PROUND — returns the numeric expression rounded to the specified power-of-ten position.

Index	Page
System 35/45 Mainframe Statements and Functions	1-5
ROM Statements Common to the System 35/45	
I/O	5-6
Mass Storage	6
Advanced Programming	6
Assembly Language	6-7
Data Communications	7
Plotter	11
Unique System 45 ROM Statements	
Graphics	12
Color Graphics	12
Data Base Management	12-13
Unique System 35 ROM Statements	
Structured Programming	13

RND — generates a pseudo random number greater than or equal to zero and less than one.

SGN — returns a 1 if the expression is positive, 0 if it is zero and -1 if it is negative.

SIN — returns the sine of the angle which is represented by the numeric expression.

SQR — returns the square root of a non-negative numeric expression.

TAN — returns the tangent of the angle which is represented by the expression.

Mass Storage Functions

TYP — returns a value which indicates what type of data will be accessed next in the specified file.

String Functions

CHR\$ — converts a numeric value between -32 768 and 32 767 into a string character.

ERRM\$ — returns a string equal to the system error message.

KBD\$ — returns a string containing all the key codes since last use of this function. (Found in I/O ROM for System 35.)

LEN — returns the current character length of the string expression.

LWC\$ — returns a string with all uppercase letters converted to lowercase.

NUM — returns the decimal equivalent of the 8-bit binary value of the first character of the string expression.

POS — determines the position of a substring within a string and returns the character position of the first character of the second string within the first or 0 if it is not present.

REV\$ — returns a string containing the elements of the specified string with the order of the characters reversed.

RPT\$ — causes the specified string expression to be repeated the specified number of times.

TRIM\$ — deletes any leading or trailing blanks from the string expression.

UPC\$ — returns a string with all lowercase letters converted to uppercase.

VAL — returns the numeric value, including any exponent, of a string of digits so that the value can be used in calculations.

VAL\$ — returns a string representing the numeric expression in current output mode.

Matrix Functions

COL — returns the number of columns (rightmost subscript) in the working size of the specified numeric or string array.

DET — returns the determinant of the specified numeric matrix of the last numeric matrix which was inverted if no matrix is specified.

DOT — returns the inner product of two vectors.

ROW — returns the number of rows (second subscript from right) in the working size of the specified numeric or string array.

SUM — returns the sum of all the elements in a numeric array.

Print Functions

LIN — used with the PRINT statement and causes a carriage return and the specified number of line feeds to be output.

PAGE — used with the PRINT statement and causes a form feed to be output.

SPA — used with PRINT and DISP to output a specified number of blank spaces.

TAB — used with PRINT and DISP and causes the next item to be output beginning in the specified character position.

Statements

General Statements

AUTO — allows lines to be numbered automatically as they are entered.

BEEP — outputs an audible beep.

COM — dimensions and reserves memory space for simple and array variables in a "common" memory area.

CONT — continue execution of the program.

DATA — provides constants and quoted or unquoted text from which READ and MAT READ obtain values for numeric and string variables.

DEFAULT ON — makes numeric overflow errors non-fatal.

DEFAULT OFF — makes numeric overflow errors fatal.

DEG — used to set degree mode for results and arguments of trigonometric function.

DEL — delete program lines specified.

DIM — used to declare the number of dimensions and the maximum number of elements in each dimension for real precision array variables and initializes all elements to zero.

DISABLE — deactivates ON KEY, ON KBD and ON INT.

DISP — outputs items on the display lines of the CRT.

EDIT — allows a string value to be altered.

EDIT LINE — allows a stored BASIC line to be altered.

ENABLE — activates ON KEY, ON KBD and ON INT.

END — terminates program execution.

FIXED — sets fixed mode for output of numeric values and specifies the number of digits to the right of the decimal point on the range zero through eleven.

FLOAT — sets floating point mode (scientific notation) for output of numeric values and specifies the number of digits to the right of the decimal point on the range zero through eleven.

FOR/NEXT — defines how many times a FOR-NEXT loop is to be executed.

GOSUB — transfers program control to the subroutine.

GOTO — transfers program control to the specified line.

GRAD — used to set grad mode for all results and arguments of trigonometric functions.

IF . . THEN — provides conditional branching.

IMAGE — specifies the format which directs the output of the PRINT USING statement.

INPUT — allows values to be assigned to variables from the keyboard during program execution.

INTEGER — used to dimension and reserve storage space for integer precision variables.

LET — used to assign a value to a variable or variables.

LINPUT — allows any combination of characters to be assigned to a string variable during program execution.

LIST — output program on standard printer.

LIST KEY — output Special Function Key definition on standard printer.

ON ERROR — declares a branching that is to occur when an error is detected.

ON . . GOSUB — allows any one of one or more subroutines in the same program segment to be accessed based on the value of the numeric expression.

ON. .GOTO — allows program control to be transferred to one of one or more statements in the same program segment based on the value of the numeric expression.

OFF ERROR — cancels any ON ERROR condition currently active.

OPTION BASE — allows the default lower bound of arrays to be specified as one rather than zero.

OVERLAP — sets the computer to the overlapped processing mode allowing computation and multiple I/O operations to occur simultaneously.

PAUSE — suspends program execution.

PRINT — outputs the items specified in the print list to the standard printer.

PRINT USING — allows the exact form of printed output to be determined by the image format string.

PRINT ALL IS — defines the standard print all device used when PRT ALL is latched.

PRINTER IS — used to define the standard printer for the system.

RAD — used to set radian mode for all results and arguments of trigonometric functions.

RANDOMIZE — reevaluates the random number seed.

READ — specifies variables for which values are to be assigned from a DATA statement.

READY # — allows I/O to a device to resume after it was suspended because of an I/O error on that select code.

REAL — used to dimension and reserve storage space for non-subscripted and array variables and declare them as full precision.

REDIM — allows a new working size for an array to be defined.

REM — allows insertion of non-executable remarks into the listing of a program to provide documentation and make the program easier to follow.

REN — repositions the DATA pointer to the beginning of the specified DATA statement.

RETURN — statement with no expression is the last line in a subroutine and transfers control back to the line following the GOSUB statement. Statement with expression supplies value for multiple-line DEF FN.

RUN — begin execution of the program.

SCRATCH — used to erase program, variables, keys, common to all R/W memory.

SERIAL — cancels the effect of any previous OVERLAP statement and sends the computer to the serial processing mode.

SECURE — prevents selected lines or an entire program from being listed.

SHORT — used to dimension and reserve storage space for simple and array variables and declare them as short precision.

STANDARD — sets standard mode for output of numeric values.

STOP — terminates program execution and sets the program pointer to the lowest numbered line.

WAIT — causes program execution to be delayed the approximate number of milliseconds before it continues.

Subprogram Statements

CALL — transfers control to a subroutine subprogram.

COM — dimensions and reserves memory space for simple and array variables in a "common" memory area.

DEF FN — defines a single- or multiple-line function.

FN END — is the last line in a multiple-line function subprogram.

SUB — is the first line of a subroutine subprogram.

SUB END — is the last line in a subroutine subprogram and transfers control back to the calling program.

SUB EXIT — used to transfer control from a subroutine subprogram back to the calling program before SUB END is executed.

RETURN X — returns values from functions.

Array Statements

MAT. .copy — result array = operand array — copies the value of each element in one array to a second array.

MAT. .function — result array = function array — causes each element in the operand numeric array to be evaluated by the specified system function.

MAT. .multiplication — result matrix = operand matrix₁ * operand matrix₂ — multiplies two matrices together.

MAT. .initialize — array variable = numeric expression — assigns the value of the expression to every element in an array.

MAT. .scalar operation — result array = scalar operator operand array — allows an arithmetic or relational operation to be performed on each element of an array using a constant scalar. The following operators are allowed: +, -, *, /, =, <>, <, >, >=, <=.

MAT. .operation — result array = operand array operator operand array — allows an arithmetic or relational operation to be performed on corresponding elements of two arrays. The following operators are allowed: +, -, ., /, =, <>, >, <, >=, <=.

MAT. .INV — establishes a square matrix as the inverse of the specified square matrix.

MAT. .TRN — establishes a matrix as the transpose of a specified matrix.

MAT. .IDN — establishes an identify matrix = all elements equal zero except the main diagonal which all equal one.

MAT. .CON — sets all elements in a numeric array to one.

MAT. .ZER — sets all elements in a numeric array to zero.

MAT. .CSUM — finds the sums of the elements of the columns of a numeric matrix and stores them in a vector.

MAT. .RSUM — finds the sums of the elements of the rows of numeric matrix and stores the sums in a vector.

MAT. .INPUT — allows values to be assigned from the keyboard to the elements of an array during program execution.

MAT PRINT # — records all elements of the specified arrays on to a mass storage medium.

MAT PRINT — causes the specified arrays to be printed on the standard printer.

MAT READ — specifies that values for all elements in an array or arrays are to be read from a DATA statement(s).

MAT READ # — reads values for the elements of the specified arrays from a mass storage medium.

Keyboard Statements

OFF KEY # — deactivates a corresponding ON KEY # statement.

ON KEY # — allows any Special Function Key to be used for program control.

OFF KBD — deactivates a corresponding ON KBD statement. (Found in I/O ROM on System 35.)

ON KBD — allows the keyboard to be used as a peripheral. (Found in I/O ROM on System 35.)

RESUME INTERACTIVE — re-enables live keyboard capability previously disabled with SUSPEND INTERACTIVE.

SUSPEND INTERACTIVE — disables live, interactive keyboard operations while a program is running.

TYPEWRITER ON — allows TYPWTR key to be “pressed” from within a program and puts the keyboard in typewriter mode.

TYPEWRITER OFF — disables a previous TYPEWRITER ON statement and returns the keyboard to normal mode.

Mass Storage Statements

ASSIGN — used to open a data file by assigning a number to it.

BUFFER # — attaches a 256 byte, semi-permanent buffer from user Read/Write memory to the specified file.

CAT — prints information about user files.

CAT TO — puts information about user files into a string array. (Found in AP ROM on System 35.)

CHECK READ — turns on verify after write.

CHECK READ OFF — deactivates a previous corresponding CHECK READ statement.

COPY — copies a file from one location to a previously undefined file.

CREATE — establishes a data file of the specified size and places an EOF mark in the first word of every record.

GET — loads into memory a string data file or a program saved with the SAVE statement.

INITIALIZE — enables an unused mass storage medium to be used by establishing physical records and main and spare directories.

LINK — loads into memory a program or any string data file without destroying the values of variables.

LOAD — puts back into memory a program stored with the STORE statement, destroying any lines in memory.

LOAD ALL — loads entire contents of user R/W memory.

LOAD BIN — loads the specified binary file into memory without altering any other binary routines already in memory.

LOAD KEY — loads Special Function Key definitions from a file stored with STORE KEY.

MASS STORAGE IS — specifies the standard (default) mass storage device.

OFF END # — deactivates a corresponding ON END # statement.

ON END # — declares a branching that is to occur when an EOR or EOF mark is encountered during a PRINT # or READ # operation.

PRINT# — records values onto the specified file.

PROTECT — guards a file against accidental erasure.

PURGE — erases the specified file from the storage medium.

READ# — retrieves values for variables from the specified file.

RENAME — allows any file to be given a new name.

RE-SAVE — allows a program to be written into a file that had been created with SAVE without purging the file first.

RE-STORE — allows a program to be written into a file that had been created with STORE without purging the file first.

REWIND — rewinds the tape to its beginning.

SAVE — lists and records all or some of program lines in memory into data file.

STORE — stores all program lines and binary routines in memory into a program file on the specified mass storage device.

STORE BIN — stores into a special file all user binary programs in memory.

STORE ALL — stores into a special file the entire contents of user Read/Write memory with the exception of the files table.

STORE KEY — stores all Special Function key typing aid definitions into a special key file.

Debugging Statements

TRACE — traces program logic flow in all or part of program.

TRACE ALL — traces all program logic flow and variable assignments.

TRACE ALL VARIABLES — monitors value changes of all variables in a specified program segment or throughout entire program.

TRACE PAUSE — used as a breakpoint causing execution to halt before a specified line is executed a certain number of times.

TRACE VARIABLE — monitors value changes of selected variables.

TRACE WAIT — causes the computer to wait a specified amount of time after each line which causes a trace printout.

NORMAL — cancels all tracing operations.

CRT Escape Sequence Commands

Escape sequence commands allow programs to control form-filling and other editing functions on the CRT. They can be used to program changes in selective parts of the display without affecting the rest. A program can request an operator to enter additional information and then position that information anywhere on the CRT.

To execute an escape sequence, the command is PRINT CHR\$(27)& followed by the symbols given below.

SYMBOL	FUNCTION
“A”	Move cursor up one row
“B”	Move cursor down one row
“C”	Move cursor right one column
“D”	Move cursor left one column
“H”	Move cursor to first row of CRT memory, first column
“F”	Move cursor to row after last row of CRT memory, first column
“S”	Roll printout up one line (like “up” arrow key)
“T”	Roll printout down one line (like “down” arrow key)
“1”	Set tab stop at cursor’s column
“2”	Clear tab at cursor’s column
“3”	Clear all tabs
“J”	Clear screen from cursor position
“K”	Clear line from cursor
“L”	Insert a blank line before cursor line
“M”	Delete cursor line & close up gap
“P”	Delete character at cursor
“Q”	Turn on insert character mode; insert to left of cursor
“R”	Turn off insert-character mode

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

- "Y" Disables control codes
- "Z" Enables control codes
- "E" Reset the CRT
- "I" Freeze all lines above cursor line
- "m" Unfreeze the lines which were previously frozen
- "&awrnC" Set cursor position to specified row (w) & column (n) CRT memory
- "&awynC" Set cursor position to specified row (w) & column (n) printout area

Highlight Escapes

To access the CRT special features (blinking, inverse video and underline), the command is PRINT CHR\$(27)&"&d"& followed by these symbols:

- @ Clear all special features
- A Set blinking
- B Set inverse video
- C Set inverse video and blinking
- D Set underline
- E Set underline and blinking
- F Set underline and inverse video
- G Set underline, inverse video and blinking

System 45C Color Features

- H White
- I Red
- J Yellow
- K Green
- L Cyan
- M Blue
- N Magenta
- O Black



ROM Statements Common to the System 35/45

The System 35 or 45 provides for the addition of optional ROM packages. Statements found in the I/O, Mass Storage, Data Comm, Advanced Programming, Assembly Language and Programming ROMs are common to both the System 35 and System 45. These ROMs enhance the BASIC language operating system by providing the following additional statements.

I/O ROM Statements/Functions Data Transfer Operations

COLLECT — forces buffer reclamation during simultaneous ENTER and OUTPUT operations.

CONVERT — allows I/O data to be converted from one coded representation to another. This could be used, for example, to convert between ASCII code and EBCDIC.

ENTER — transfer data from an external device into a specified set of program variables.

IMAGE — specifies a format to be used with an ENTER or OUTPUT statement.

OUTPUT — transfers the contents of a specified data list to an external device.

READ BIN — inputs a single byte (8 bits) or word (16 bits) from an external device.

TRL — transfer limit character (stop byte/word) for terminating an ENTER.

WRITE BIN — outputs a single byte (8 bits) or word (16 bits) to an external device.

Bit Manipulation Functions

These functions allow the program to manipulate general 16-bit binary values in order to test or form special bit patterns. The logic operations are performed on a bit-by-bit basis.

BINAND — binary AND of two values.

BINCMP — binary one's complement of one value.

BINEOR — binary exclusive OR of two values.

BINIOR — binary inclusive OR of two values.

BIT — true/false indication for specified bits.

DECIMAL — octal to decimal base conversion.

LASTBIT — returns the state of the last bit shifted or rotated out of the word specified in the ROTATE or SHIFT functions.

OCTAL — decimal to octal base conversion.

ROTATE — left/right end-around rotate of a value by a specified number of bits.

SHIFT — left/right shift of a value by a specified number of bits.

Direct Register Access

These statements and functions allow the program to directly address the interface cards for special operations and protocols not provided for

by other high-level, BASIC language statements.

IOFLAG — tests the interface flag (ready/busy) line.

IOSTATUS — tests the interface status line.

READIO — direct input from a specified interface register.

WRITEIO — direct output to a specified interface register.

WAIT READ — read register after card flag line set.

WAIT WRITE — write (output) to a specified register after card flag line set.

Interrupt Operations

Interrupts for the purpose of data transfers are handled automatically by the OUTPUT and ENTER statements. To service interrupts for other purposes, the user may write BASIC language service routines.

CARD ENABLE — enables an interface to interrupt the system when a condition specified by the CONTROL MASK statement is met.

CONTROL MASK — specifies the conditions for which an interface is allowed to generate an interrupt to the system.

OFF INT# — cancels a previous ON INT# specification for the given select code.

ON INT# — specifies the location of a BASIC language service routine for each select code, with 15 levels of user-assigned priority.

SYSTEM TIMEOUT OFF — deactivates the System Timeout Message/Beep.

SYSTEM TIMEOUT ON — turns on the System Timeout Message/Beep.

Asynchronous Data Communications Operations

The following statements allow asynchronous data communications when used with the Terminal Emulator program (BASIC language).

TBUF\$ — reads data input from buffer, setup by TOPEN. (System 45 B/C only.)

TCLOSE — disables TOPEN. (System 45 B/C only.)

TDISP — allows character (string) output to the keyboard entry area of CRT. (System 45 B/C only.)

TOPEN — enables branching in the resident BASIC program when a character is input from the 98036A (RS-232-C) interface. (System 45 B/C only.)

HP-IB Control

Normal data transfers between the System 35 or 45 and HP-IB devices are done using the OUTPUT and ENTER statements. HP-IB command bytes always have the 8th bit as odd parity. The following statements are used to access the extended control features of the HP-IB.

ABORTIO — issues the interface-clear bus message.

CONFIGURE — allows the System as active controller to set up a talker and listener(s) for a data transfer in which it will not participate.

EOI — sends a byte with EOI line indicating the end-of-data message.

LOCAL — clears remote enable or sends the go-to-local bus message.

LOCAL LOCKOUT — disables devices from front panel control.

PASS CONTROL # — passes active control to another device on the HP-IB.

PPOLL — conducts a parallel poll operation.

PPOLL CONFIGURE — configures bus devices for parallel poll response.

PPOLL UNCONFIGURE — unconfigures bus devices for parallel poll response.

REMOTE — sets the remote enable bus control line.

REQUEST — requests service from the active controller.

RESET — issues the universal or selective device-clear message.

SENBUS — allows direct control of the HP-IB. Any bus message may be sent with the attention (ATN) line true or false.

STATUS — reads status information from an interface card or conducts a serial poll operation with an HP-IB device.

TRIGGER — issues the group execute trigger universal command.

Mass Storage ROM

The Mass Storage ROM provides drivers to communicate with external disc drives.

Most of the Mass Storage statements and commands are included in the mainframe. Additional statements are:

FCREATE — creates a binary data file.

FPRINT — stores an array into a binary data file.

FREAD — reads an array from a binary data file.

Advanced Programming ROM

The Advanced Programming ROM provides statements which improve the data manipulation capabilities of the System 35/45.

Data Manipulation

LEX — compares two strings and indicates which comes first according to the current specified lexical order.

LEXICAL ORDER IS — allows the user to select the collating sequence for string MAT SORT commands.

LWC\$ — transforms uppercase characters to lowercase equivalents.

MAT REORDER — rearranges an array into the physical order corresponding to the SORTed order.

MAT SEARCH — returns information about current conditions (value locations, minimum/maximum values, etc.) in an array.

MAT SORT — sorts array data in ascending, descending or lexical (alphabetical) fashion.

UPC\$ — transforms lowercase characters to uppercase equivalents.

Assembly Language Execution & Development ROM and Assembly Language Execution ROM

The System 35/45 Assembly Language System offers the experienced programmer complete control of the System 35 or System 45 CPU(s) through the use of machine intructions, pseudo-instructions, and extensions to the BASIC language.

The Execution and Development ROM provide the following BASIC statements and functions, as well as those listed for the Execution ROM.

ISOURCE — allows the programmer to write Assembly Language source statements that are integrated within the framework of a BASIC language program.

IASSEMBLE — reduces source code to object code.

IBREAK — allows breaking (pausing) at either a data or program location; allows transfer at break time to a BASIC subprogram and then resumption of the Assembly subprogram; provides 8 independent break points.

INORMAL — discontinues conditions set up by IBREAK.

IPAUSE ON/OFF — allows or disallows STEP and PAUSE to operate normally within an assembled

routine; when STEpping through a routine, if the source is present, the Assembly Language instruction and associated comment is displayed.

IDUMP — allows printing of memory location in any of 5 formats: binary, octal, decimal, hexadecimal and ASCII.

ICHANGE — changes the content of a memory location to a specified value.

IADR — returns the value of a symbol, usually an address.

IMEM — returns the content of a memory location.

OCTAL — converts a decimal expression to its octal image.

DECIMAL — converts an octal expression to its decimal image.

The Execution ROMs support these BASIC statements:

ICOM — sets aside special Read/Write area in memory to accept the output of the assembler.

ICALL — transfers processor control to an assembled subprogram; allows the passing of parameters between BASIC programs and Assembly subprograms.

ISTORE — allows storage of object modules on mass storage devices.

ILOAD — allows retrieval of object modules from mass storage devices.

IDDELETE — deletes selected Assembly Language modules.

ON/OFF INT# — establishes or discontinues end of line branch condition for ISRs.

Utilities. A number of utilities have been provided in the System to help make programming tasks easier and to give you direct access to some of the operating system's capabilities and routines. Such capabilities as basic arithmetic operations between full precision numbers, conversions between number types, storage and retrieval of string and numeric variables, mass memory Read/Write record and unformatted printing are provided in utility form.

ASSEMBLY LANGUAGE INSTRUCTIONS

Here is a listing of mnemonics, a brief description of the functions and typical execution times in microseconds (number in parenthesis).

Load/Store Functions

LDA (LDB) — load register A (or B) with the content of a specified location (specified by an operand, the

syntax of which is not detailed here). (2.2)

STA (STB) — store content of register A (or B) in specified location. (2.2)

CLR — clear the specified number of words, beginning at the location pointed at by the A register. (6.7)

XFR — transfer the specified number of words, from the location starting at the address pointed at by the A register to the location pointed at by the B register. (11.5)

Integer Math Functions

ADA (ADB) — add the content of the specified location to register A (or B). (2.2)

TAC (TCB) — perform a two's complement of the A (or B) register. (1.5)

MPY — integer multiply. (13.5)

Branch Functions

JMP — jump to specified location. (1.3)

JSM — subroutine jump to specified location. (2.2)

RET — return from subroutine. (2.7)

Test Branch Functions

CPA (CPB) — compare A (or B) to the content of the specified location, skip if unequal. (2.7)

SZA (SZB) — skip to specified location if register A (or B) is 0. (2.3)

RZA (RZB) — skip to specified location if register A (or B) is not 0. (2.3)

SIA (SIB) — skip to specified location if register A (or B) is 0; then increment register A (or B) by 1. (2.3)

RIA (RIB) — skip to specified location if register A (or B) is not 0; then increment register A (or B) by 1. (2.3)

Test/Alter/Branch Functions

ISZ (DSZ) — increment (decrement) content of specified location, skip if new content is 0. (3.2)

SAP (SBP) — skip to specified location if the A (or B) register is positive, i.e. bit 15 is 0. (2.3)

SAM (SBM) — skip to specified location if the A (or B) register is negative, i.e. bit 15 is 1. (2.3)

SLA (SLB) — skip to specified location if the least significant bit of register A (or B) is 0. (2.3)

RLA (RLB) — skip to specified location if the least significant bit of register A (or B) is 1. (2.3)

SOC (SOS) — skip to specified location if Overflow is clear (or set). (2.3)

SEC (SES) — skip to specified location if Extend is clear (or set). (2.3)

Shift/Rotate Functions

SAL (SAR) — shift the A register left (or right) the indicated number of bits with all vacated positions becoming 0. (2.8)

SBL (SBR) — analogous to SAL (SAR) using B register. (2.8)

AAR (ABR) — shift the A (or B) register right the indicated number of bits with the sign bit filling all vacated bit positions. (2.8)

RAL (RAR) — rotate the A register left (or right) the indicated number of bits; bit 15 will rotate into bit 0 (left shift) or bit 0 will rotate into bit 15 (right shift). (2.8)

RBL (RBR) — rotates the B register analogous to RAL (RAR). (2.8)

Logical Functions

AND — logical AND between register A and specified location, result in register A. (2.2)

IOR — inclusive OR between A register and specified location, result in register A. (2.2)

CMA (CMB) — perform a one's complement of the A (or B) register. (1.5)

Stack Functions

PWC (PWD) — push the specified register (full word) onto the stack pointed at by the C (or D) register. (3.8)

PBC (PDB) — push the lower byte (right half) of the specified register onto the stack pointed at by the C (or D) register. (3.8)

WWC (WWD) — withdraw a full word from the stack pointed at by the C (or D) register and place it in the specified register. (3.8)

WBC (WBD) — withdraw a byte from the stack pointed at by the C (or D) register and place it in the lower byte (right half) of the specified register.

CBL (CBU) — clear (or set) the Cb register (C and Cb together act as a 17-bit address for PBC and WBC). (2.0)

DBL (DBU) — clear (or set) the Db register. (2.0)

BCD Math Functions

MRX (MRY) — mantissa right shift on Ar1 (or Ar2), a special BCD floating point machine register. (11.0 for MRX, 6.2 for MRY)

MLY — mantissa left shift on Ar2 for one digit. (5.3)

DRS — mantissa right shift of Ar1 for one digit. (9.3)

NRM — normalize the Ar2 mantissa. (4.8)

CMX (CMY) — ten's complement of Ar1 (9.8 or Ar2. (3.8)

FXA — fixed point addition; the mantissa of Ar1 and Ar2 are added together and the result placed in Ar2. (6.7)

MWA — mantissa word addition of B to Ar2. (4.7)

FMP — fast BCD multiply. (24.3)

FDV — fast BCD divide. (23.5)

CDC — clear decimal carry. (1.8)

SCD (SDS) — skip to specified location if decimal carry is clear (or set). (2.3)

I/O Functions

SFC (SFS) — skip to specified location if I/O Flag line is false (or true). (2.3)

SSC (SSS) — skip to specified location if I/O Status line is clear (or set). (2.3)

EIR (DIR) — enable (disable) the interrupt system. (2.0)

SDI (SDO) — set DMA inwards (or outwards); reads from peripheral (or memory), writes to memory (or peripheral). (2.0)

DMA — enable the DMA mode. (2.0)

DDR — cancel the DMA instruction. (2.0)

Miscellaneous Functions

NOP — null operation. (2.2)

EXE — execute the contents of any of the first 32 registers; the operand specifies which register. (1.3)

Basic Data Comm ROM

The 9845/9835 Data Comm ROMs are designed to provide flexible BASIC language statements that cover many variations on a data communications protocol. This was accomplished by the use of "secondary keywords" that define parameters needed by the BASIC "keyword." All secondary keywords have default values that are used if the secondary keyword is not specified in the BASIC statement. Therefore, you need include only the secondary keywords whose values you want changed from the default. It is not necessary to specify keywords if all the defaults are correct.

Note that some keywords are unique to either the Basic Data Comm ROM or the RJE Bisync ROM and some are common to both.

Basic Data Comm	RJE Bisync	KEYWORD	SECONDARY KEYWORD	Definition
x	x	CCOM		Reserves memory to be used for Data Comm buffers.
x	x	CCONNECT		Configures and establishes the Data Comm line connection.
x	x		EXTERNAL	Selects external clocking for transmit and receive.
x	x		HANDSHAKE OFF	Directs the HP 98046A Interface card to ignore modem handshake lines (used for hardwired connections.) If not specified, the 98046 will continue to monitor modem handshake lines.
x	x		INSPEED = /OUTSPEED =	Sets the transmit and receive speeds independently. Used for split speed operations.
x	x		LOST CARRIER	Provides an auto-disconnect from the Data Comm lines if certain RS-232-C signals remain false for a specified time period.
x	x		NO ACTIVITY	Provides an auto-disconnect from the Data Comm line if no data is transmitted or received for a specified time period.
x	x		SPEED =	Sets the baud rate for transmit and receive.
x	x	CCONTROL		Provides control features for several aspects of the Data Comm channel. At least one secondary keyword must be included.
x			ACK OFF	Disables ENQ/ACK handshaking.
x			ACK ON	Enables ENQ/ACK handshaking. Used to prevent overruns at high speeds.
x	x		BREAK	Sends a 500 millisecond break to the Data Comm channel in async. Sends EOT for RJE Bisync.
x	x		DRIVER 1 ON/OFF DRIVER 2 ON/OFF DRIVER 3 ON/OFF	Provides direct control from BASIC over 3 RS-232-C lines. Useful for implementing asynchronous Half-Duplex line protocols.
	x		FORMAT = NONE FIXED COMP TRUNC BOTH	Selects the type of data compression and/or truncation to be used for automatic data blocking.
x	x		INTMASK =	Specifies one or more of nine conditions to generate an interrupt to BASIC.
	x		RECORD SIZE =	Specifies the record size to be used for automatic data blocking.
x	x		RESET	Aborts all Data Comm activity on the channel. Clears buffers.
	x		RETRIES =	Specifies the number of retransmissions for a block which receives a negative acknowledgement.

Basic Data Comm	RJE Bisync	KEYWORD	SECONDARY KEYWORD	Definition
x			READALL ON	Permits all characters, including control characters, to be passed to a BASIC program.
x	x		SUSPEND	Suspends activity on the data channel. Does not clear buffers.
	x		TEXT = ASCII BINARY	Specifies whether data is to be converted to ASCII before storing in string variables.
	x		TRANSPARENT/NORMAL	Specifies the mode of data transmission.
x			XON	Enables XON/XOFF handshaking.
x	x	CDISCONNECT		Terminates all Data Comm activity for a specified channel.
x	x		HOLD	Terminates all data comm activity, but maintains all modem control lines at their current state.
x	x	CDUMP		Dumps the Trace buffer in format specified by the secondary keyword.
x	x		CODE HEX	Dumps the trace information in the format of hexadecimal numbers.
x	x		CODE OCT	Dumps the trace information in the format of octal numbers.
x	x		FORCE	Dumps the trace information regardless of the status of the data comm channel.
x	x		LTRACE ALL	Dumps all trace information to the PRINTER IS device.
x	x		LTRACE CTL	Dumps the trace of the control bytes sent to the 98046 driver.
x	x		LTRACE IN	Dumps the input line trace to the PRINTER IS device.
x	x		LTRACE OUT	Dumps the output line trace to the PRINTER IS device.
x	x		LTRACE TX	Dumps the transmitted data trace to the PRINTER IS device.
x	x		STRING =	Dumps the trace information to the specified string variable rather than the PRINTER IS device.
x		CMODEL ASYNC		Configures the desktop computer and Data Comm Interface for asynchronous communication.
x			ALERTN =	Specifies the maximum number of characters to read before generating input.
x			CHARLENGTH =	Defines the number of bits per character.
x			CHECK =	Defines the type of parity to be used. 4 No parity 3 Odd parity 2 even parity 1 always 1 0 always 0
x			GAP =	Specifies the delay in milliseconds between successively transmitted characters. Useful for writing character mode terminal emulators.



Basic Data Comm	RJE Bisync	KEYWORD	SECONDARY KEYWORD	Definition
x			HALF DUPLEX/ FULL DUPLEX	Specifies the Data Comm channel type to be used.
x			INBUFFER =	Specifies the amount of CCOM to allocate for the input buffer.
x			INSEP =	Defines the separator to be used to delimit lines of received data.
x			MEMLIMIT =	Specifies the amount of CCOM to allocate to the transmit and receive queues.
x			OUTSEP =	Defines the separator(s) to be used to delimit lines of received data.
x			PROMPT =	Defines the character(s) to be interpreted as a prompt from the remote computer.
			STOP BITS = 1 1.5 2	Specifies number of stop bits.
x			TBUFFER =	Specifies the amount of CCOM to allocate to the trace buffer.
x			WORDLENGTH =	Defines the number of bits per character.
	x	CMODEL 2780/ CMODEL 3780		Configures the desktop computer and Data Comm interface for either IBM 2780 or 3780 emulation.
	x		EBCDIC/USASCII	Specifies the code to be used on the Data Comm channel.
	x		INBUFFER =	Specifies the amount of CCOM to allocate for the input buffer.
	x		HALF DUPLEX/FULL DUPLEX	Specifies the Data Comm channel type to be used.
	x		MAXRPB =	Specifies the maximum number of records that will be transmitted in a single block.
	x		MEMLIMIT =	Specifies the amount of CCOM to allocate for the transmit and receive queues.
	x		PRIMARY/SECONDARY	Specifies whether the desktop computer is to emulate a primary or secondary station.
	x		TBUFFER =	Specifies the amount of CCOM to allocate to the trace buffer.
	x		MSV2	Selects Siemens MSV2 protocol.
x	x	CREAD		Inputs data from the Data Comm channel to a string variable.
x	x	CSTATUS		Places Data Comm status information in a 16-element integer vector.
x	x	CSTAT		Returns the value of a specified element of the CSTATUS vector.
x	x	CTRACE		Specifies the type of tracing to be performed on the Data Comm channel.
x	x		IN ON/IN OFF	Starts or stops tracing of data input from the channel.
x	x		OUT ON/OUT OFF	Starts or stops tracing of data output from the desktop computer to the Data Comm interface.

Basic Data Comm	RJE Bisync	KEYWORD	SECONDARY KEYWORD	Definition
x	x		STOP ERROR	Sets the desktop computer to stop tracing on occurrence of a Data Comm error.
x	x		STOP FULL	Sets the desktop computer to stop tracing when the trace buffer is full.
x	x		TX ON/TX OFF	Starts or stops tracing of data output to the Data Comm channel.
x	x		WRAP	Sets the desktop computer to trace continuously by treating the trace buffer as a circular buffer.
x	x	CWRITE		Transfers data from the desktop computer to the Data Comm channel.
	x		ENDBLOCK	Terminates a block with the ETB character and sends it to the Data Comm channel. Used to override automatic data blocking.
x	x		ENDLINE	For Async: sends the OUTSEP character to the Data Comm channel. For RJE Bisync: terminates a record.
	x		ENDTEXT	Terminates a block with the ETX character and sends it to the Data Comm channel. Used to override automatic data blocking.
x	x		EOT	For async: sets the request to send line low. Useful for managing Half-Duplex lines. For RJE Bisync: Sends the EOT character and idles the Data Comm channel.
x	x		ON INT#	Specifies the location of a BASIC language service routine for each select code, with 15 levels of user-assigned priority.
x	x	OFF INT#		Cancels a previous ON INT# for the given select code.

Plotter/Graphics ROM

The Plotter ROM provides the statements necessary to create graphics images on the 9872A or 7245A plotters when interfaced to the System 35. These same statements are also found in the System 45 Graphics ROM.

PLOTTER IS — selects and initializes plotter.

CLIP/UNCLIP — limits the area in which lines will be drawn.

GCLEAR — advances paper.

LIMIT — defines the physical limits of the plotting area.

LOCATE — defines the plotting area rectangle.

MSCALE — sets millimeters as plotting units.

SCALE — defines the range of X and Y in user units.

SETGU/SETUU — set graphics units/user units.

SHOW — specifies a rectangle in user units to be expanded to fit plotting rectangle.

Plotting Statements

PDIR — rotates relocatable coordinate system.

DRAW — specifies absolute draw.

ILOT — specifies incremental plot.

LINETYPE — selects a dash pattern.

MOVE — specifies absolute move.

PEN — selects pen.

PENUP — lifts pen.

PLOT — specifies absolute plot.

RLOT — specifies relocatable plot.

Axis and Labeling Statements

AXES — draws linear axes.

CSIZE — sets character size.

FRAME — outlines plotting area.

GRID — draws linear grid.

LABEL/LABEL USING — draws a label.

LDIR — sets label direction.

LORG — sets label origin mode.

Interactive Statements

CURSOR — reads cursor position.

DIGITIZE — reads cursor with wait.

POINTER — sets cursor position.

Unique System 45 ROM Statements

The System 45 Graphics ROM provides all the statements found in the System 35 Plotter ROM with the addition of statements necessary to create graphics images on the System 45's CRT as well as other peripheral plotters. These commands make it easy to get sophisticated data representation in a hurry.

Graphics Control Statements

DUMP GRAPHICS — creates hard-copy of CRT graphics on optional internal printer.

GLOAD — loads graphics memory.

GSTORE — stores graphics memory.

GRAPHICS/EXIT GRAPHICS — selects graphics/alpha mode.

PLOTTER IS ON/OFF — activates/deactivates the specified plotting device.

System 45C

Graphics Firmware

The System 45C Graphics firmware, built into the mainframe, include all the statements of the Plotter ROM and Graphics ROM plus an additional set of 30 high-powered statements that expand your use of color and the graphic input devices such as the light pen and 9874 Digitizer.

ALPHA — enables the CRT alphanumeric raster so that items specified by the **DISP**, **LABEL KEY #**, **PRINT**, and **PRINT USING** statements directed to the CRT are visible on the CRT.

AREA COLOR — selects the fill color to be used whenever **MAT PLOT**, **MAT IPLOT**, **MAT RPLOT**, **MAT SYMBOL**, **POLYGON** or **RECTANGLE** are used with **FILL** specifier.

AREA INTENSITY — selects the fill color to be used whenever **MAT PLOT**, **MAT IPLOT**, **MAT RPLOT**, **MAT SYMBOL** **POLYGON** or **RECTANGLE** are used with a **FILL** specifier.

CONVERGE — allows you to adjust the electron beams to cause them to strike the proper dot locations on the CRT.

DEGAUSS — used to remove the effects of residual magnetism on the CRT.

EXIT ALPHA — disables the CRT's alphanumeric raster.

GRAPHICS INPUT IS — selects (and presets conditions for) the active graphics input device. This statement also turns off any other active graphics input device.

GRAPHICS INPUT IS Array — specifies an nx3 array as the active graphics input device.

GRAPHICS INPUT. . .IS OFF and **IS ON** — deactivates/enables a previously specified graphics input device.

GSTAT — returns information concerning the most recently activated graphics input and output devices.

KEY LABELS — allows you to save the text written for the soft keys by storing them into a string variable (maximum length = 480 characters).

LABEL KEY # — labels the prompt in print area lines 27 and 28 for each of the individual softkeys located on the CRT's bezel.

LABEL KEYS — labels all eight of the prompts for the softkeys.

LAXES — draws axes for a plot with labeled major tick marks.

MAT AILOT — quickly plots the X-Y data contained in the first two columns of an array.

MAT APLOT — quickly plots the X-Y data contained in the first two columns of an array.

MAT ARPLOT — quickly plots the X-Y data contained in the first two columns of an array.

MAT IPLOT — allows the contents of an array to be plotted incrementally as X-Y coordinate pairs with pen control.

MAT PLOT — plots the contents of an array.

MAT RPLOT — plots the contents of an array as relative X-Y coordinate data with pen control.

MAT SYMBOL — allows labeling with user-defined characters.

MEMORY — specifies that all the plotted points in a memory plane are to be displayed in one color.

OFFGKEY — disables the end-of-line branch set by the **ON GKEY** statement. Once the **OFF GKEY** statement has been executed, you must re-execute the **ON GKEY** statement for the graphics input device to cause the end-of-line branch.

ON GKEY — sets a prioritized end-of-line branch for interactive graphics.

PLOTTER IS Array — specifies an array as an active plotter.

POLYGON — draws a regular polygon (which can approximate a circle or arc) and/or alternatively fills the interior of the polygon with the fill color specified by the most recently executed **AREA COLOR** or **AREA INTENSITY** statement.

RECTANGLE — draws and/or alternatively fills a rectangle with the color currently specified by **AREA COLOR** or **AREA INTENSITY**.

TRACK. . .IS OFF — disables the tracking marker on the plotters.

TRACK. . .IS ON — places a marker on the graphics output device which tracks the movement of the cursor on the graphics input device.

WHERE — reads pen position.

IMAGE/45 Data Base Management ROMs

The IMAGE/45 ROMs provide the statements and functions necessary to create and manipulate data within a data base on the System 45.

IMAGE/45 Statements

DBBACKUP — creates a backed-up version of all or part of the data base.

DBCLOSE — closes the data base or writes all changed root file information to a disc.

DBCREATE — creates the data set files on the volume specified in the schema.

DBDELETE — deletes the current record of the specified set (the last entry involved in a **DBGET** or **DBPUT**). The current record is unchanged.

DBERASE — erases all data entries and related path information from one or more of the sets in a data base.

DBFIND — used with detail data sets to locate and set the current record pointer to the first entry in the chain which has the key item value identified by the argument parameters. This enables a chained **DBGET** to be performed.

DBGET — reads an entry from the specified set and puts it into the data base buffer string.

DBINFO — obtains information about data items, sets, paths and volumes.

DBOPEN — opens data base for access.

DBPURGE — purges the root file and all or some of the data files of the specified data base.

DBPUT — adds an entry to the specified manual master or detail data set.

DBRECOVER — un-backs up a data base.

DBUPDATE — updates the current record for the specified set (the last entry involved in a DBGET or DBPUT).

DEL FN — deletes a function subprogram from memory.

DEL SUB — deletes a subroutine subprogram from memory.

LOAD SUB — loads one or more subprograms into memory from the specified program file, appending them to the current program.

MSI — abbreviation for MASS STORAGE IS.

PACK USING — transfers data from program variables into the data base buffer string according to the format of the corresponding PACK FMT statement. Can be used before DBPUT or DBUPDATE.

PACK FMT — specifies a format to be used by PACK and UNPACK on the buffer string.

UNPACK USING — transfers data from the buffer string to variables in the pack list. Can be used after DBGET or DBINFO.

PRINT LABEL — writes a label onto the disc in either the current default mass storage device or the device specified after ON.

READ LABEL — reads the label off the current default mass storage device or off the device specified after ON.

VOLUME DEVICES ARE — sets up the volume device table so it contains the specified msus's and their associated labels.

IMAGE/45 Functions

DEV\$ — returns a string whose value is the current default mass storage device.

HOLE — returns the largest number of contiguous file records on the current default mass storage device.

IOR — performs a bit-by-bit inclusive OR operation on two integer expressions.

Unique System 35 ROM Statements

Structured Programming ROM

The System 35's Structured Programming ROM provides you with the looping and decision-making capabilities of PASCAL while retaining the friendly interpretive features of BASIC. This enhancement allows you to organize your programs more logically and facilitates program documentation, modification and maintenance.

IF. . . THEN. . . ELSE. . . END IF — causes up to two choices of action to be implemented as the result of a conditional test.

INDENT — causes logical indentation of your program code.

LOOP. . . EXIT IF. . . END LOOP — repeats the statements in a structured loop as long as the EXIT IF condition(s) is false.

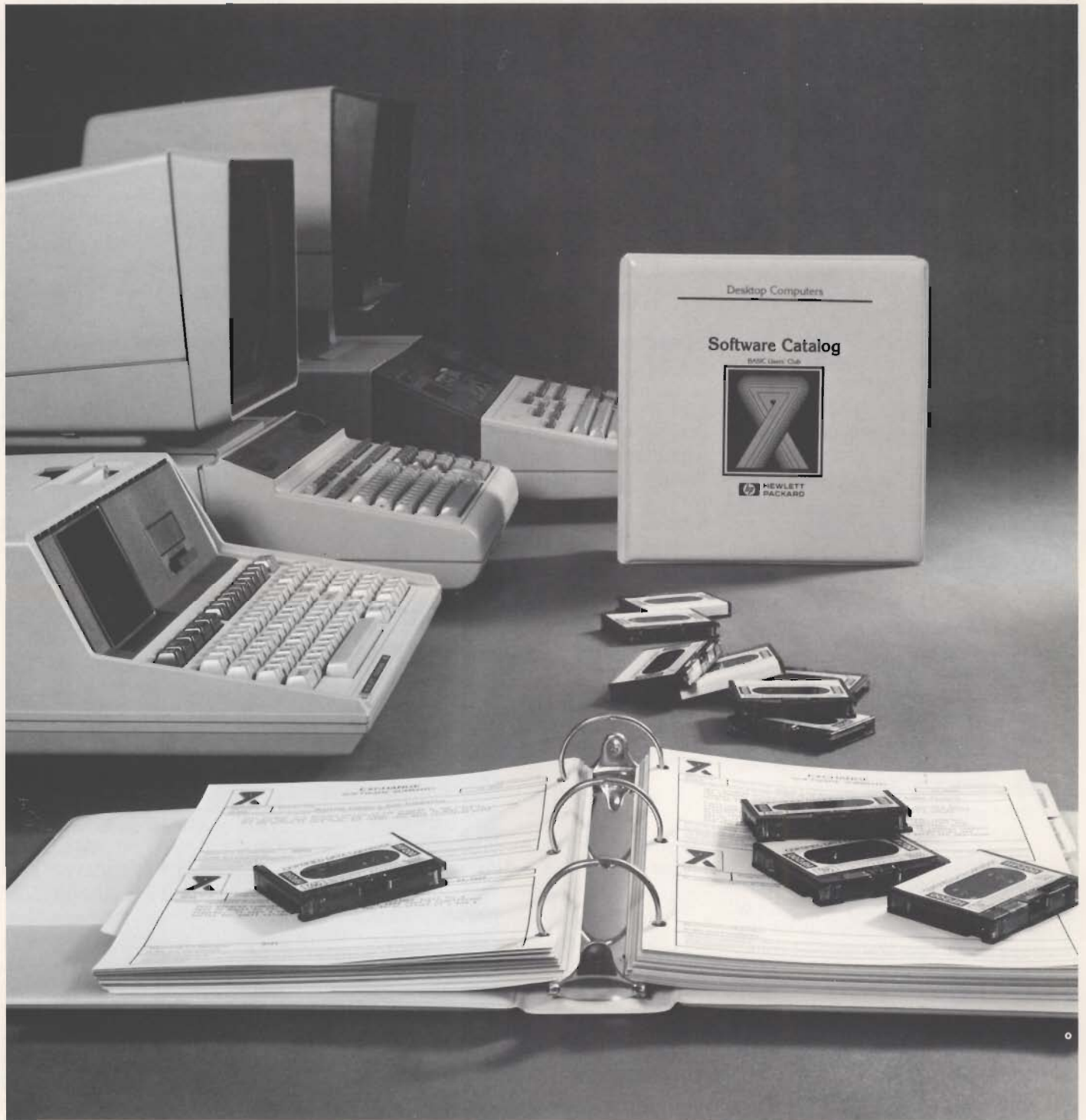
REPEAT. . . UNTIL — repeats the statements in a structured loop until the condition(s) in the UNTIL statement is true.

SELECT. . . CASE. . . CASE ELSE. . . END SELECT — provides the execution of a choice of action depending upon the result of a conditional test.

WHILE. . . END WHILE — repeats a structured loop when its condition is true.

XREF — prints a listing of the identifiers and where they occur (line numbers) in your program.

Software Catalog Subscription Service



Before you invest time writing your own programs, you may want to consider the Software Catalog. This catalog serves as a single reference source for all BASIC software available from Hewlett-Packard, third-party suppliers and exchange programs submitted to the BASIC Users' Club. For more information contact:

Hewlett-Packard
Software Catalog
Subscription Service
P. O. Box 590
Ft. Collins, Colorado 80521

For assistance call the HP regional office nearest you: Eastern 301/258-2000, Midwest 312/255-9800, Southern 404/955-1500, Western 213/877-1282, Canadian 416/678-9430. Or write to Hewlett-Packard, 3404 East Harmony Road, Fort Collins, Colorado 80525; in Europe, Hewlett-Packard GmbH, Desktop Computer Division, Herrenberger Strasse 110, D-703 Boeblingen, Postfach 1430, West Germany; elsewhere in the world, Hewlett-Packard Intercontinental, 3495 Deer Creek Road, Palo Alto, California 94304.

