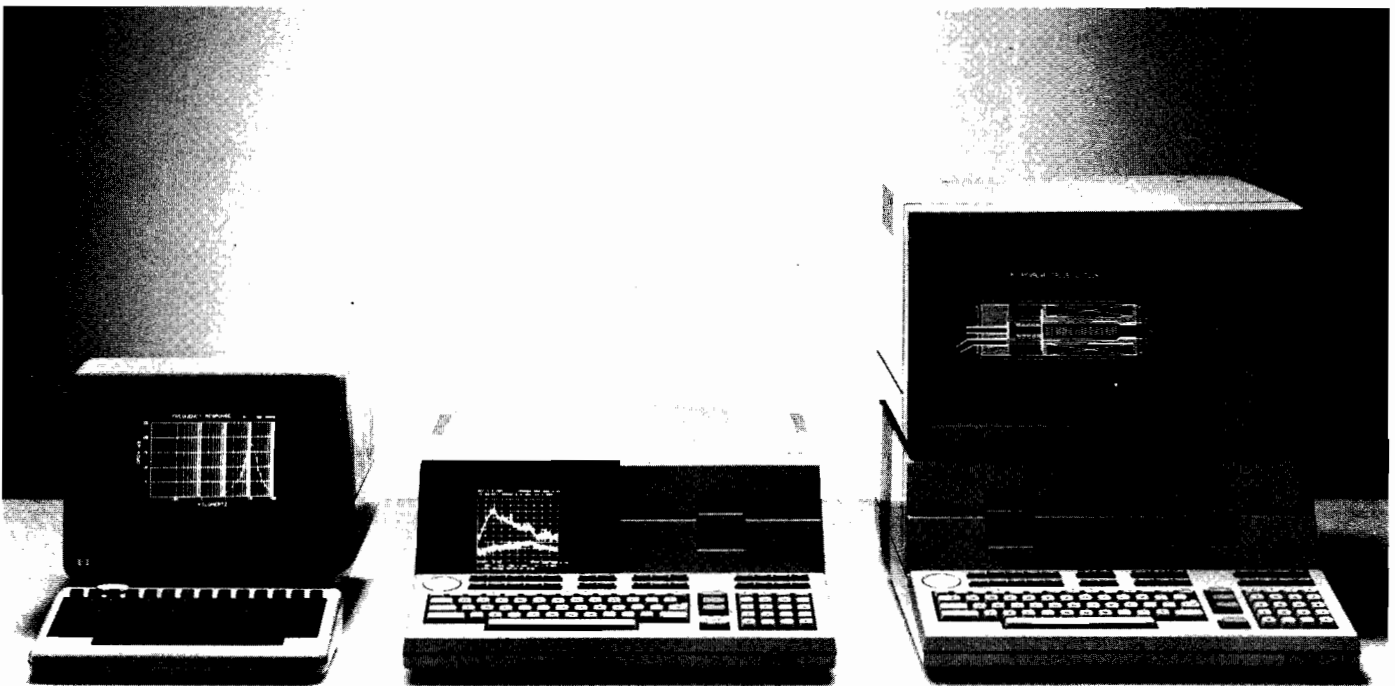


HP Series 200 Models 216, 226 and 236 HPL 2.0 and 2.1 Language System



Technical Data*, January 1984



Introduction

Hewlett-Packard's Series 200 Personal Technical Computers are based on the MC68000, a powerful 16-bit Motorola microprocessor with 32-bit internal architecture and 8 MHz clock rate. HPL supports three of the Series 200 computers – Models 216, 226 and 236.

The smallest, Model 216, fits easily on a desk. It features a 9-inch (229 mm) CRT, small, detached keyboard and memory from 128K to 768K bytes. A variety of disc drives, printers and other peripherals are available to convert it into a stand-alone system. It features both HP-IB[†] and RS-232-C built-in interfaces. The Model 216 is a "personal" technical computer.

The Model 226 features a 7-inch (175 mm) CRT, built-in 5¼-inch flexible disc drive with 264K bytes capacity and up to 2 megabytes memory. Having a larger base and keyboard than the Model 216, it is suited for placement on a rack with other electronic devices and instrumentation. With built-in HP-IB interface, it finds its ideal application in computer-aided test (CAT) applications.

The Model 236 features a full 12-inch (310 mm) CRT, dual, built-in 5¼-inch flexible disc drives, and up to 2 megabytes of memory. Its larger screen adapts it more to computer-aided engineering (CAE) applications such as engineering design and analysis. Like the Model 226, it features a built-in HP-IB interface. Its large mass storage capacity (528K bytes) also supports these more demanding applications.

Series 200 machines feature BASIC, HPL and Pascal, graphics, user-definable soft keys, a special "rotary control knob" for easy editing and simulations, memory-mapped I/O, prioritized interrupt, internal system clock and timers, and built-in slots for additional memory or interface cards. In addition, a special backplane expander allows up to an additional 4 megabytes of memory or eight interface cards to any Series 200 computer.

* Data subject to change.

† HP-IB is Hewlett-Packard's implementation of IEEE Standard 488-1978.

HPL 2.0 and 2.1 Capabilities

HPL, Hewlett-Packard's high-level programming language, is renowned for its high performance and efficiency in handling computational and I/O operations. The power and performance of HPL are borne out by the success of the HP 9825. HPL provides you this high performance while maintaining an easy-to-use software development environment.

HPL can be purchased as either a ROM-based or RAM-based system. The ROM-based version is HPL 2.0; the RAM-based version is HPL 2.1. (HPL 2.1 is a revision of HPL 2.0, which corrects known problems in HPL 2.0.) Programs written in HPL 2.0 will run without modification on HPL 2.1 systems; the reverse is also true. There are no plans to offer a ROM version of HPL 2.1.

All 9825 HPL features have been carried over to the Series 200 HPL Language System, allowing compatibility with existing 9825 software. This reduces the 9825 to Series 200 upgrade to a simple process.

The Series 200 HPL Language System includes support of the commands available with the 9825T, 98211A Matrix ROM and the 98228A Flexible Disc ROM. The 9825's BCD-based math is emulated on the Series 200 so all arithmetic computations will be executed under the same numeric range and precision as the 9825. Numeric results are the same, whether on a 9825 or Series 200. The 9825's interface card register maps are emulated by the HPL Language System as well. Even though the Series 200 computers have a new set of interface cards, you can be assured of software transportability, even down to most register-level HPL I/O statements.

In addition, Series 200 HPL has been enhanced. This allows you to utilize the new hardware capabilities as well as improve the inherent capability of HPL as a programming language. HPL enhancements fall into the following categories:

Human Interface

- Full screen CRT multi-line program editing
- Full screen CRT output with enhanced graphics commands
- Keyboard knob for enhanced program-editing control
- Dynamic special-function key definition and CRT labelling
- User-definable system printer select code and printer character width
- Dumping alpha and graphics displays selectively to external printing devices or I/O buffers
- International keyboards
- 10-deep recall buffer

Mass Storage

- Complete emulation of 9825 tape operations on disc-drive mass storage
- Compatibility with 98228A Flexible Disc ROM commands allowing access to 9885 & 9895 Flexible Disc Drives via resident 9825-HPL software
- Compatibility with BASIC and Pascal 2.0 or later Language System data files
- Support of binary data files for increased performance and easy string I/O buffer storage
- Support of the 82900 Series 5¼" mini-disc drives
- Optional use of DMA with the 9895 flexible disc drive for higher performance

Graphics

- Complete support for CRT and Plotter Graphics
- Support of Color Graphics on the Color Video Interface (98627A)
- Dump Graphics of CRT image to compatible printer

I/O Enhancements

- Emulation of 9825 Interface Cards' register maps
- DMA available with HP-IB interface as well as GPIO interface
- Access to transfer buffer pointers
- Aborting of active buffer transfers
- Software-alterable HP-IB address
- Interrupt-routine link cancellation
- Reading of data into substrings
- Rotary control knob for program interrupts
- Program access and control of internal clock and timers
- Transparent HP-IB transfer termination on End-or-Identify

Additional Enhancements

- Read/data command constructs for enhanced variable initialization
- Extended cross-reference to include 'p' variables
- String conversions to and from any base up to 38
- String expression execution of keyboard keycodes from a program to "push" keyboard keys
- 32 program flags for software use
- Full access to Series 200 memory space
- Powerfail recovery enhancements as an option to the Model 226/236 (not available on the Model 216)

HPL Software Transportability

Series 200 HPL provides a complete and easy upgrade path for your 9825 software and application in either of two ways: (1) direct interface or (2) via external 9885/9895 flexible-disc mass storage.

Interface Interconnect. 9825 HPL programs and data can be easily sent over HP-IB to the Series 200. For example, program transfer involves a simple 'list #731' of the program from the 9825, and a short program on the Series 200 that reads in the program and stores it into memory.

External Mass Storage. Any 9825 HPL program or data file stored on a 9885/9895 flexible disc can be read by the Series 200 with the HPL Language System. In addition, disc file transfers allow key files to be transferred from the 9825 to the Series 200. Typical HPL external mass storage commands can be used to store and load HPL programs and data.

Due to the physical differences between the 9825 and Series 200 hardware, there are several differences in the systems:

- Keyboard access returns Series 200 keycodes instead of 9825 keycodes
- The keyboard and display control register do not exist on the Series 200
- Any 98035 real-time clock dependent code is not supported and should be modified to utilize the internal clock and timers on the Series 200
- Due to the differing internal architecture, 9825 memory dump and binary files are not supported by the Series 200 HPL Language System

In addition, the 9862A Plotter is not supported on the Series 200.

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

HPL 2.0 and 2.1 Operating Characteristics

Language System Memory Requirements

ROM-based HPL 2.0

1 ROM board AND the following RAM depending on model number:

Model 216	14K bytes RAM
Model 226	11K bytes RAM
Model 236	21K bytes RAM

RAM-based HPL 2.0 and HPL 2.1

The following RAM depending on model number:

Model 216	122K bytes RAM
Model 226	119K bytes RAM
Model 236	129K bytes RAM

Range

Dynamic Range

-10^{99} to -10^{-99} , 0, 10^{-99} to 10^{99}

Internal Calculation Range

-10^{511} to -10^{-511} , 0, 10^{-511} to 10^{511}

Math Hierarchy

Highest priority	functions, flag references, r-variables ^ (exponentiation) implied multiply - (unary minus) *, /, mod +, - all relational operators (=, >, <, <=, >=, #) → not and
Lowest priority	or, xor

Operators of the same level in an expression are executed from left to right with operations within parentheses performed first.

Built-in Functions

Mathematical and trigonometric functions and operations are included in the following, with typical execution times in milliseconds.

Absolute (abs)	0.12
Fraction (frc)	0.30
Integer (int)	0.40
Maximum (max)	variable
Minimum (min)	variable
Modulus (mod)	1.66
log	19.56
ln	17.01
e^x	17.23
10^x	.21
Raise to power	39.51
Random number (rnd)	0.92
Sign (sgn)	0.12
Square root (sqr)	4.433
(square root sign)	4.433
Sine (sin)	16.39
Cosine (cos)	16.25
Tangent (tan)	10.21
Arcsine (asn)	32.39
Arccosine (acs)	32.24
Arctangent (atn)	33.78
+	0.226
-	0.309
*	0.40
÷	0.42
Power of ten round (prnd)	0.53
Digit round	0.65
Logical operators	
and	
or	
xor	
not	
Relational string and numeric operators	
=	Equal
>	Greater than
<	Less than
>= or =>	Greater than or equal to
<= or =<	Less than or equal to
# or <> or ><	Not equal to
String operator	
&	Concatenation

HPL Keyword Summary

General Functions

abs – returns the absolute value of the expression.
acs – returns the principal value of the arccosine of the expression in the current angular units.
add – returns the sum of the expressions, added in the current numeric mode, decimal or octal.
asc – returns the ASCII equivalent of the specified keycode.
asn – returns the principal value of the arcsine of the expression in the current angular units.
atn – returns the principal value of the arctangent of the expression in the current angular units.
avm – returns the size (bytes) of unused read/write memory.
band – returns the 16-bit result of ANDing the expressions.
bit – returns the binary value of the bit position in expression two indicated by expression one.
cln – returns the current program line number.
cmp – returns the 16-bit binary one's complement of the expression.
cos – returns the cosine of the expression.
cycle – returns the number of cycle interrupts that have occurred since last referenced.
drnd – returns the value of the first expression, rounded to the number of digits indicated by the second expression.
dto – returns the octal equivalent of the decimal value expressed.
emsg – returns detailed error message string.
eor – returns the 16-bit binary result of the exclusive ORing of the expressions.
erl – returns the line number where the current error occurred.
ern – returns the current error number.
exp – returns e (2.71828...) raised to the expressed power.
flg – returns flag status.
frc – returns the fractional part of the expression.
fti – rounds and changes the expression to integer precision. The result can be stored in a two-character field.
fts – changes the expression to split precision for storage in a four-character field.
knob – returns the knob's count and direction of rotation.
kstat – returns the keyboard status indicating knob, control key or shift key usage.
int – returns the integer value of the expression.
ior – returns the 16-bit result of the inclusive OR operation on the expression.
itf – returns a full-precision number from the packed, integer-precision number (a two-character string).
key – returns the earliest, unprocessed keycode in the keyboard buffer.
ln – returns the natural log (\log_e) of the expression.
log – returns the common log (\log_{10}) of the expression.
machine – returns information regarding internal configuration of the computer such as characters per line and graphics resolution.
max – returns the largest value in the list.
min – returns the smallest value in the list.
nal – returns the last program line number plus one; used with store to store strings.

otd – returns the decimal equivalent of the octal value expressed.
pi (π) – returns the value of the constant pi.
prnd – returns the first expression rounded to the power of ten indicated by the second expression.
res – returns the result of the last keyboard operation not stored in a variable.
rnd – returns a pseudo-random number from 0 to (less than) 1.
rom – returns the current error ROM identifier.
rot – returns the result of binary rotation of the 16-bit equivalent of the first expression rotated the number of bits indicated by the second expression.
sgn – returns sign of the expression.
shf – returns the result of right-shifting the 16-bit binary equivalent of the first expression, the number of places indicated by the second expression. A negative second expression shifts the byte to the left.
sin – returns the sine of the expression.
sqr ($\sqrt{\quad}$) – returns the square root of the expression.
stf – unpacks and returns a split-precision number from its four-character string.
tan – returns the tangent of the expression.
tn[^] – returns 10 raised to the specified power.
units – displays the currently-set angular units.

String Functions

cap – returns an equivalent string of uppercase characters.
char – returns the ASCII equivalent character or characters.
len – returns the character length of the string.
num – returns the ASCII-decimal value of the character.
pos – returns the character position of the second string within the first.
rtime – returns the time of the internal clock.
str – returns the ASCII character equivalent to the expression. The expression can be of any number base.
val – returns the numeric value of the string. The string can be of any number base.

General Statements

aclr – clears the CRT alpha buffer and allocates scrolling pages.
adump – dumps the contents of the CRT alpha buffer to the system printer.
aoff/aon – turns the CRT alpha buffer off or on.
aprt – prints the specified array's elements on the internal printer. The rightmost subscript cycles the fastest.
ara – performs array arithmetic operations.
beep – sounds the computer's beeper.
cfg – clears either all 32 program flags or only the specified flags.
cll – calls the subroutine having the specified label, passing the value of any optional variables as passed parameters.
cmf – complements either all 32 program flags or only the specified flags.
cont – continues program execution, either from the current point or from the specified point.
cret – return from periodic timer interrupt routine.
crt – controls display enhancements on the Model 236 such as inverse video and blinking.
csv – clears simple variables A through Z.
data – provides constants and text characters for use with the read statement.

deg – sets degrees units for angular calculations.
 del – deletes selected program lines.
 dim – reserves memory for specified variables.
 dret – return from delay timer interrupt routine.
 dsp – displays the items listed.
 end – halts program execution and sets the program counter to zero.
 enp – enters and prints data entered from the keyboard.
 ent – enters data from the keyboard.
 erase – erases either all programs and variables or specified areas.
 fetch – displays the specified program line or special-function key definition.
 find – prints line numbers of all occurrences of the specified character sequence.
 flt – sets floating point notation.
 for – defines start of a for-next loop.
 fxd – sets the fixed-point format.
 grad – sets the grads units for angular calculations.
 gsb – directs program execution to the specified subroutine.
 gto – directs program execution to the specified line.
 idn – creates identity (square) matrices. All elements are zero except major diagonal elements which are one.
 if – allows execution of the remainder of the line if the equation is true. If false, execution immediately branches to the next line. Any relational operator can be used.
 ina – initializes each element of the array to the specified value.
 inv – stores the inverse matrix of an array into another array. The matrix' determinant can optionally be calculated as well.
 jmp – directs program execution the relative number of lines forward or back.
 kloff/klon – turn special-function-key labels off or on.
 kret – returns execution to the main program after the key buffer is emptied. Also allows return from a knob interrupt routine.
 list – lists the entire program to the active printing device.
 list # – lists the entire program to an active external printing device.
 list(sfk) – lists the special-function key definition.
 listk – lists all special-function key definitions.
 lkd – disables live keyboard mode.
 lke – enables live keyboard mode.
 mat – carries out matrix multiplication.
 mdec – sets the decimal mode for binary operations.
 moct – sets the octal mode for binary operations.
 mret – allows return from match timer interrupt routine.
 next – terminates for-next loop and tests for loop completion.
 nor – clears the master program flag, either while executing all lines or only for the specified line numbers.
 on cycle – enables periodic timer interrupt routine.
 on delay – enables delay timer interrupt routine.
 on err – enables an error-trapping routine.
 on key – enables a keyboard interrupt routine.
 on knob – enables knob interrupt routine.
 on match – enables match-timer interrupt routine.
 pbeep – outputs a tone of specified frequency for a specified duration.
 pkbd – allows the execution of an ASCII string of keyboard keycodes.
 prt – prints the list of items on the active printer.
 prtsc – allows the definition of the system printer select code and printer width.
 rad – sets radians units for angular calculations.
 rdm – redimensions the array(s) to the specified dimensions.
 read – assigns values from a data statement to the variables specified.
 ret – ends a subroutine and returns program execution to the main program (line after gsb).
 rkbd – enables a remote keyboard to control the computer. Keyboard type can also be selected between ASCII and Series 200 keyboards.
 rstr – resets data pointer to the start of the specified data statement, or the first data statement if none is specified.
 run – begins program execution either at line zero or at the specified line.
 sfg – sets either all 32 program flags to one or only the specified flags.
 sfk – defines special function key and optional soft label.
 smpy – multiplies each element of an array by a scalar number.
 spc – outputs the expressed number of line feeds on the internal printer.
 stime – sets the time of the internal clock.
 store – stores program lines from an executing program.
 stop – stops program execution either immediately or, optionally, at the specified line.
 sysboot – exits HPL and brings in a new language system.
 tabxy – moves print position to the specified CRT alpha buffer coordinates.
 trc – sets the master flag and, optionally, trace flags for specified program lines.
 trn – transposes rows and columns between two arrays.
 wait – causes the program to wait for the specified time in milliseconds.
 xref – prints a cross reference of program variables and line numbers, using the current program in memory.
 % – allows storing text without syntax checking.

I/O Functions

bred – returns the contents of the specified, active, interrupt buffer.
 iof – returns interface flag state.
 ios – returns interface status.
 pol – conducts a parallel poll on the HP-IB.
 rdb – returns one 16-bit binary character code from the specified device.
 rdi – returns a status byte from the interface specified by wti0.
 rds – returns the current status word from a specified interface or serial poll.
 rss – returns the 98036/98626 interface status register byte.

I/O Statements

buf – sets up and names a data buffer of type read/write of the specified type interrupt, fast read/write or DMA.
 cli – sends the abort message to all devices on the HP-IB.
 clr – sends the clear message, either to all devices or to only a selected device by including the device address in the select code.

cmd – sends the string of data characters to the specified HP-IB device.

conv – sets up a conversion table referenced by red and wrt statements.

ctbl – sets up a conversion table.

dev – assigns a name for use in place of the select code in I/O operations.

eir – enables an interrupt from the specified select code.

eol – specifies up to seven optional ASCII characters from an end-of-line sequence for wrt operations. An optional delay can also be specified.

equ – equates the ASCII character string with the name, for use with cmd.

fnt – sets up a list of format specs for red and wrt operations.

iret – ends an interrupt service routine and returns to main program.

lcl – sends the local message to all HP-IB devices or, if the select code includes a device address, sends a clear local-lockout message.

llo – sends the local-lockout message to all HP-IB devices.

oni – references an interrupt service routine associated with the peripheral's select code. Disables link if no reference is indicated.

par – sets the parity type used for I/O checking.

pct – passes active control to the specified HP-IB device.

polc – sets parallel poll bits on the specified HP-IB device.

polu – clears parallel poll bits on the specified device.

red – reads and stores data from the specified device. Also allows reading data into substrings.

rem – sends the remote message to either all HP-IB devices or only one device when its address is included in the select code.

rqc – requests service from the HP-IB system controller and sends the serial status byte upon response to a serial poll.

trf – transfers data between an I/O buffer and a peripheral device. The total number of bytes to transfer and the terminating character can be optionally specified.

time – causes an I/O operation to wait for a device to become ready for the specified number of milliseconds.

trg – sends the trigger message to the specified HP-IB device.

wrt – outputs the items to the specified device.

wsc – outputs a control word to the specified interface.

wsm – outputs a mode word and, optionally, a control word to the specified 98036/98626 interface.

wtb – outputs the byte representing each number or character to the specified device.

wtc – outputs a control byte to the specified interface. Also can reset HP-IB interface to a new bus address or just reset the interface.

wti0 – specifies an interface for successive wti or rdi operations.

wti – output a control byte to a specified interface register.

Mass Storage Functions

dtype – returns a code indicating the type of drive, disc and data format at the default disc address.

type – returns the next item-type in a disc data file.

Mass Storage Statements

asgn – assigns a number to an existing disc file name and indicates optional drive number and a return status variable.

avd – disables automatic 'tape' file verification.

ave – enables automatic 'tape' file verification.

cat – prints a catalog of files on the specified disc or default drive.

chain – loads a program from the specified disc file without erasing existing program variables.

copy – duplicates the contents of the source disc onto the destination disc. Can also be used to only copy a specified number of records.

drive – sets the default unit and, optionally, the select code for successive disc operations.

ert – erases the current emulated tape track, beginning with the specified file.

fdf – positions the emulated tape at the specified file on the current track.

files – assigns numbers to up to 10 disc files.

get – loads the program from the specified disc file.

getb – loads the specified disc binary program file.

getk – loads the special-function-keys disc file at any time.

idf – provides information on the current 'tape' file.

init – runs the disc initialization routine at the specified interleave factor and loads bootstraps.

kill – erases the specified disc file.

killall – erases all disc user files.

ldb – loads a binary program from the specified 'tape' file.

ldf – loads the specified 'tape' file into the appropriate area of memory or load data from the specified 'tape' file into variables.

ldk – loads the special-function-key file into memory at any time.

ldp – loads and executes a program from a 'tape' file.

mrk – marks the number of files, beginning at the emulated tape's current position.

msi – specifies the current system mass storage device.

on end – enables a branch to the specified line or label when a disc EOF or EOR mark is encountered during read and write operations.

open – creates a disc data file of the specified size.

rcb – records the resident binary program onto a 'tape' file.

rcf – records a program or program segment onto a 'tape' file. The program can optionally be secured from listing or displaying as well. Variables can be recorded onto a 'tape' file as well.

rck – records the special-function-key definitions onto a 'tape' file.

renm – renames a disc file.

repk – repacks files on the disc.

resave – stores a program (or only the specified lines) in an existing disc.

rew – rewinds the emulated tape.

rprt – prints the list of data items on the disc file, starting at the specified record.

read – reads data from the disc file, starting at the specified record.

save – stores either the entire program on the disc file or only the specified block of lines.

saveb – stores the resident binary program on the specified disc file.

savek – stores all special-function-key definitions on the disc file.
 sprt – prints the list of data items on the disc file.
 sread – reads data from a disc file.
 tlist – catalogs 'tape' files on the active printer.
 trk – specifies the emulated tape track (0 or 1) for successive operations.
 vfy – verifies the contents of a 'tape' file against the original in memory.
 voff – disables the verification mode.
 von – enables the verification mode.

Graphics Statements

bplt – allows plotting any series of dots on the CRT or external color video display.
 cplt – moves the pen the specified distance away from the current point.
 csiz – specifies the size, shape and lettering direction for label statements.
 dig – reads, computes and stores the current pen position in user units for an external plotter.
 gclr – clears the display graphics buffer.
 gdump – dumps the display graphics buffer to an external raster dump printer.
 gload – loads the display graphics buffer from a string.
 goff/gon – turns off or on the display graphics buffer.
 gptr – draws a graphics cursor at the specified point on the display.
 gstore – stores the display graphics buffer to a string.
 iplt – moves the pen the number of X and Y units from its current position.
 lbl – prints characters on the current graphics device.
 lim – restricts plotter pen movement to the stated bounds in user units.
 line – specifies the type of line plotted with graphics commands to an external plotter.
 ofs – offsets the origin.
 pclr – sets default graphics values except scale units, select code, P1, P2, pen location and pen#.
 pen – raises the plotting pen.
 pen# – selects the plotting pen. Also allows for pen off, erase, exclusive or, or normal for use with display graphics.
 plt – moves the plotting pen to the specified X,Y point and will optionally raise or lower the pen before or after the move.
 psc – sets the select code for all plotting operations.
 ptyp – sets a graphics lettering mode.
 scl – locates the origin and specifies user units for plotting operations.
 xax – draws an X axis with optional tic marks and labels.
 yax – draws a Y axis with optional tic marks and labels.

Powerfail Functions*

power – returns 0 if the ac power source fails; returns 1 if ac power is up; returns -1 if no Powerfail hardware is present.
 ptime – can return either the amount of battery time used up during a power failure or the duration of a power failure.

* These statements require the Powerfail option (Model 226 or 236 with Opt. 050 or 98270F only). Not available on Model 216.

Powerfail Statements*

fret – return from powerfail interrupt.
 on pfail – enables a powerfail interrupt routine.
 pshutdown – turns the computer off as if the on/off switch were toggled to off.
 stime – transfers the time from the clock in the powerfail unit to the internal clock of the computer.

* These statements require the Powerfail option (Model 226 or 236 with Opt. 050 or 98270F only). Not available on Model 216.

Series 200 HPL Interface Capabilities

In addition to the built-in HP-IB Interface,* there is a choice of external interface cards and internal memory enhancements:

- 2-channel DMA Controller (98620A)
- GPIO (98622A)
- BCD (98623A)
- HP-IB (98624A)
- Serial (98626A)
- Color Video (98627A)

* Serial interface is also built into the Model 216.

DMA Controller Card

The 98620A DMA Controller Card enhances the Series 200's interfacing capability by providing two DMA channels for I/O data transfers. This high-speed I/O capability works with the 98622A GPIO, 98624A HP-IB, and internal HP-IB interfaces.

GPIO Interface

The 98622A GPIO Interface provides 16 bits of latched input and output data for bidirectional transfer of information. Extended control and status lines are available for applications that require more than one signal from the computer. Several handshake modes are also available to permit interfacing to a variety of equipment.

BCD Interface

The 98623A BCD Interface connects the Series 200 computer with bit-parallel, digit-parallel, binary-coded decimal devices for data input. Up to eight significant BCD digits, two sign bits (mantissa and exponent), exponent digit, function code digit, and an overload bit can be read. Input format is selectable, allowing two independent instruments to be read from one 98623A Interface Card. Data can also be accepted as five input bytes of pure binary information. Eight data output lines are also provided for use as general purpose control and/or data output.

HP-IB Interface

In addition to the standard built-in HP-IB Interface, there is an optional external 98624A HP-IB Interface. Both interfaces implement the IEEE 488-1980 Standard Digital Interface for Programmable Instrumentation. Both interfaces can communicate with as many as 14 HP-IB compatible instruments, connected with a maximum of 20 meters (65.6 ft.) of cable.

Serial Interface

The 98626A Serial Interface provides bit-serial communication between the Series 200 computer and asynchronous EIA RS-232-C (CCITT V.28/V.24) devices. Data rates range from 50 to 19200 baud (bits/sec.). A variety of cabling options allow for current loop, modem and terminal connections. Series 200 Terminal Emulator Software takes advantage of this interface for communication to other computers. The Serial Interface is built into the Model 216 computer.

Color Video Interface

The 98627A Color Video Interface provides the interconnection to an external color monitor. This interface connects to a high-performance, high-resolution color monitor via three outputs – Red, Green/synch, and Blue (RGB). The capabilities provided by this interface make it appear as a "soft plotter."

Transfer Rates

These are the maximum data transfer rates for the I/O cards offered on Series 200 computers. The transfer rates quoted are the maximum rates that can be attained with the type of data transfer specified. Any delay generated by the peripheral or by additional program statements or options will cause the actual data transfer rate to be lower.

– All transfer rates are in K bytes/second –

Language	HPL				
	Type of Xfer*	Int	Fast R/W	DMAr	DMAb
I/O Card					
98622A					
input	8				
output			100	480	670
98623A					
input	7.3 [†]		21 [†]		
98624A					
input	6		42	330	
output	6		70	230	
98626A [‡]					

* Abbreviations:
 Xfer – Transfer
 Int – Interrupt
 Fast R/W – Fast Read/Write
 DMAr – Direct Memory Access, regular mode
 DMAb – Direct Memory Access, burst mode
[†] 7.3K bytes/second is 450 readings/sec.
 21K bytes/second is 1.3K readings/sec.
[‡] The 98626A supports the following baud rates:

50	75	110	134.5
150	200	300	600
1200	1800	2400	3600
4800	7200	9600	19200