

HP 82401A

HP-IL Interface

Quick Reference Guide

Contents

Standard Terms	1
Keyword Summary	2
Reference Tables	23
Flag Summary	29
Error Summary	30
Keyword Index	Inside Back Cover

The simplified syntax shown in this guide uses the following conventions:

- item / item* Indicates that only one of several items should be included.
- [*item*] Indicates that an item is optional—it may be included or omitted.

Standard Terms

The following terms may be used with many of the keywords provided by the HP-IL interface. They're used to define an HP-IL device or to define a file in an HP-IL device.

device specifier:

[:] % accessory ID [(number)] / [:] assign code /
[:] device ID [(number)] / [:] device word [(number)] /
[:] HP-IL address / [.] volume label

Most mass storage statements require the : (or .) shown as optional.

device word:

DISPLAY / GPIO / GRAPHIC / HPIB / INSTRMT /
INTRFCE / MASSMEM / MODEM / PRINTER / RS232 /
TAPE

file specifier:

file name device specifier

Device specifier must begin with : (or .).

loop:

[:] LOOP

Keyword Summary

The following pages describe each keyword provided by the HP-IL interface. Refer to the owner's manual for more complete descriptions of syntax and restrictions.

Each keyword entry also defines the keyword as a *statement* or as a *function*, and states whether it operates primarily while the HP-71 is a controller, a device, or both.

All keywords are programmable.

A

ASSIGN # *channel number* **TO** *file specifier*

Statement: Associates an I/O channel number with a file and opens the file. (Controller.)

ASSIGN # 1 TO DATAFILE:TAPE

Associates channel 1 with file **DATAFILE** located in the first mass storage device and opens the file.

ASSIGN #50 TO NOTES:3

Associates channel 50 with file **NOTES** on device at address 3 and opens the file.

ASSIGN #(C-3) TO FILE1.VOLUM1

Associates computed channel number with file **FILE1** on medium with volume label **VOLUM1**.

ASSIGN #4 TO *

Closes the file associated with channel 4 and cancels that association.

ASSIGN IO *assign code list*

Statement: Associates assign codes with HP-IL devices. (Controller.)

ASSIGN IO “:TV,:TA,:PR”

Sets up specified assign codes for first three HP-IL devices.

ASSIGN IO A\$

Sets up assign codes specified by string variable.

ASSIGN IO *

Cancels all assign codes.

B**BINAND(integer , integer)**

Function: Returns the value of the binary AND operation performed on two integers. (Controller and device.)

A=BINAND(Y,Z)

Sets **A** to the value of an AND operation performed on **Y** and **Z**.

X=BINAND(X,2^N−1)

Sets **X** to the value of its last **N** bits.

IF N THEN N=BINAND(C,7)

If **N** is true, sets **N** to value corresponding to last three bits of **C**.

BINCMP(integer)

Function: Returns the value of the binary (1's) complement of an integer. (Controller and device.)

C1=BINCMP(C)

Sets **C1** to the value of the complement of **C**.

DISP BINCMP(A DIV 2)

Shifts the binary representation of **A** one bit to the right and displays its complement.

BINEOR(integer , integer)

Function: Returns the value of the binary exclusive-OR operation performed on two integers. (Controller and device.)

IF M=1 THEN A=BINEOR(Y,Z)

If **M** = 1, sets **A** to the value of the exclusive-OR operation performed on **Y** and **Z**.

Y=BINEOR(X,2^N-1)

Sets **Y** to the value of **N**-bit complement of **X**.

BINIOR(*integer* , *integer*)

Function: Returns the value of the binary inclusive-OR operation performed on two integers. (Controller and device.)

IF BINIOR(Y,Z)=255 THEN 410

Branches to line 410 if the value of the inclusive-OR operation performed on **Y** and **Z** is 255 (last eight bits set).

X=BINIOR(X,8)

Sets the fourth from last bit in **X** to 1.

BIT(*integer* , *bit position*)

Function: Returns the value of one bit of an integer. (Controller and device.)

B=BIT(R,B1)

Sets **B** to the value of bit **B1** in **R**.

IF BIT(R,3) THEN R=R-2^3

If bit 3 is set in **R**, changes **R** so that bit 3 isn't set.

IF BIT(S,6) THEN GOSUB "SRQ"

If bit 6 is set in **S**, branches to subroutine labelled **SRQ**.

C

CAT *file specifier / device specifier*

Statement: Gives a catalog of file information. (Controller.)

CAT PHONE:TAPE(3)

Gives catalog entry of file **PHONE** in third mass storage device.

CAT .VOLUM1

Gives catalog of mass storage medium having volume label **VOLUM1**.

CAT\$(*file number* , *device specifier*)

Function: Returns a string containing catalog information for a file. (Controller.)

F\$=CAT\$(2,":MASSMEM")

Sets **F\$** equal to the catalog string for the second file in the first mass storage device.

DISP CAT\$(X, ".DATA")

Displays the catalog string for file number **X** on the mass storage medium with volume label **DATA**.

CHAIN *file specifier*

Statement: Purges current file, copies specified file into main RAM, and begins executing that file. (Controller and device.)

CHAIN PRGM2:TAPE

Purges current file, loads and runs **PRGM2** from first mass storage device.

CLEAR [*device specifier* / *loop*]

Statement: Clears an individual HP-IL device or all HP-IL devices. (Controller.)

IF X=2 THEN CLEAR ":DISPLAY(2)"

If **X** = 2, clears second display device.

IF A THEN CLEAR LOOP

If **A** is true, clears all devices.

CONTROL OFF / ON

Statement: Sets the controller status of the HP-71, either causing the HP-71 to operate as the HP-IL controller or causing it to operate as an HP-IL device (*not* as the controller). (Controller and device.)

CONTROL OFF

Sets the HP-71 to immediately start operating as an HP-IL device.

IF X=20 THEN CONTROL ON

If **X** = 20, sets the HP-71 to immediately assume control of the HP-IL system.

COPY *file specifier / device specifier / loop*

[**TO** *file specifier / device specifier / loop*]

COPY [*file specifier / device specifier / loop*]

TO *file specifier / device specifier / loop*

Statement: Copies a file from one location to another.
(Controller and device.)

COPY START:TAPE(2)

Copies file **START** on second mass storage device to new file **START** in main RAM.

COPY TO BACKUP.MASTER

Copies current file to file **BACKUP** on medium with volume label **MASTER**.

COPY PRGM1:CA TO PRGM2

Copies file **PRGM1** in device “:CA” to new file **PRGM2** in main RAM.

COPY TO :MASSMEM

Copies current file to first mass storage device.

COPY FLIM:2 TO FLAM:3

Creates file **FLAM** in device at address 3, a duplicate of file **FLIM** in device at address 2.

COPY DATA:TAPE TO :INTRFCE

Copies file **DATA** in first mass storage device to first interface device.

COPY “RPT” TO :LOOP

Sends the contents of file **RPT** in main RAM to HP-IL, but not to a particular device. (Useful when operating as HP-IL device.)

CREATE DATA / SDATA / TEXT *file specifier* , *file size* [, *record length*]

Statement: Creates a data file (DATA, SDATA, or TEXT).
(Controller.)

CREATE TEXT “FILE6:1”,500

Creates TEXT file **FILE6** with 500 bytes in device at address 1.

CREATE SDATA F\$,40

Creates SDATA file with name and device specified by **F\$**, and 40 “registers” long.

CREATE DATA RIM.SHOT,10,50

Creates DATA file **RIM** with ten 50-byte records on medium with volume label **SHOT**.

Input Parameters

File Type	File Size	Record Length
DATA	Number of logical records.	Number of bytes.
SDATA	Number of “registers” (8 bytes/register).	Ignored—always 8 bytes.
TEXT	Number of bytes.	Ignored.

D

DEVADDR(*device specifier* **)**

Function: Returns the address of a device. Returns the value **−1** if the device isn’t found. (Controller.)

C=DEVADDR(D\$)

Sets **C** to address of device specified by **D\$**.

X=DEVADDR(“PRINTER(2)”) @ IF X#−1 THEN OUTPUT X;A,B

Sets **X** to address of second printer device, then sends that device the values of **A** and **B**.

P=DEVADDR(“HP82164A”) @ IF P#−1 THEN ENTER P;X\$

Sets **P** to address of HP-IL/RS-232 interface (device ID of “HP82164A”), then fetches data from that device and stores it in **X\$**.

T=DEVADDR(“%16”) @ IF T#−1 THEN COPY FILE1 TO :T

Sets **T** to address of first device with accessory ID of 16, then copies file **FILE1** to that device.

DEVAID(*device specifier* **)**

Function: Returns the accessory ID of a device. Returns the value -1 if there's no response. (Controller.)

DISP I;DEVAID(I)

Displays the value of **I** and the accessory ID of the device at address **I**.

T=DEVAID("PRINTER")

Sets **T** to the accessory ID of the first printer device.

Accessory ID Ranges

Accessory ID Range	Device Class	HP-71 Device Word
0-15	Controller	—
16-31	Mass Storage Device	MASSMEM
32-47	Printer	PRINTER
48-63	Display	DISPLAY
64-79	Interface Device	INTRFCE
80-95	Electronic Instrument	INSTRMT
96-111	Graphics Device	GRAPHIC
112-127	Analytical Instrument	—

DEVID\$(*device specifier* **)**

Function: Returns a string containing the device ID of a device. Returns a null string if there's no response. (Controller.)

DISP I;DEVID\$(I)

Displays the value of **I** and the device ID of the device at address **I**.

D\$=DEVID\$("PRINTER")

Sets **D\$** to the device ID of the first printer device.

DISPLAY IS *device specifier*

Statement: Assigns one HP-IL device to be the display device. (Controller.)

DISPLAY IS PRINTER(2)

Assigns the second printer device to be the **DISPLAY IS** device.

DISPLAY IS DISPLAY

Assigns the first display-type device to be the **DISPLAY IS** device.

DISPLAY IS *

Cancels the **DISPLAY IS** device.

E

ENABLE INTR *interrupt mask*

Statement: Specifies the events that can cause an HP-IL interrupt. (Controller and device.)

ENABLE INTR L+2^N

Enables an interrupt to occur only for events specified by the value of the expression.

IF E THEN ENABLE INTR 16

If **E** is true, enables an interrupt to occur only when the HP-71 becomes an active talker device (mask value 16).

Interrupt Mask

Bit	Decimal Value	Interrupt Event	Controller	Device
7	128	Interface Clear	X	X
6	64	Listener	(X)	X
5	32	Controller		X
4	16	Talker	(X)	X
3	8	Service Request	X	
2	4	Device Clear	X	X
1	2	Trigger	X	X
0	1	Device Dependent	X	X

ENTER *device specifier / loop* [**USING** *image reference*]
[; *variable list*]

Statement: Reads data from HP-IL into numeric and string variables. (Controller and device.)

ENTER 2; N,A\$

Enters from device at address 2 two values, a number stored in **N** and a string stored in **A\$**.

ENTER 3 USING 1000; X,Z(1)

Enters from device at address 3 two numeric values stored in **X** and **Z(1)**. Uses formatting from **IMAGE** statement at line 1000.

ENTER INTRFCE(2); A1,A2

Enters from second interface device two numeric values, stored in **A1** and **A2**.

ENTER "HP82164A" USING "80A"; X\$,Y\$

Enters two strings from first HP-IL/RS-232 interface (device ID "HP82164A") and stores them in **X\$** and **Y\$**. Each string has 80 characters.

ENTER 3 USING "#,B"; I

Enters from device at address 3 one byte, whose value is stored in **I**.

ENTER LOOP; B1\$

Enters from HP-IL (device not specified) a string, which is stored in **B1\$**.

Statement terminates after all variables assigned values *and* line feed character or End Byte message received (or immediately if line feed not required).

Free-field input uses sequential numeric characters up to next non-numeric character for numeric variable assignment, and uses all characters until string fills, until line feed received, or until carriage return and line feed received for string variable assignment.

Formatted input uses characters according to specified image symbols (refer to "Reference Tables").

Flag —23 set affects input only while the HP-71 is controller. Basically, End Of Transmission messages replace line feed characters and End Byte messages as variable and statement terminators.

I

INITIALIZE [*volume label*] *device specifier* [, *directory size*]

Statement: Initializes a mass storage medium (sets the volume label, creates the directory, and clears the medium). (Controller.)

INITIALIZE :TAPE(2)

Initializes the medium in the second mass storage device, giving it a blank volume label and default directory size.

INITIALIZE TEST1:A,55

Initializes the medium in the device at address **A**, giving it a volume label of “TEST1” and a 55-entry directory.

INITIALIZE A\$,35

Initializes the medium in the device specified in **A\$**, giving it a volume label also specified in **A\$** and a 35-entry directory.

L

LIST IO

Statement: Lists all defined assign codes and associated HP-IL addresses. (Controller and device.)

LIST IO

Lists all active assign codes.

IF L THEN LIST IO

If **L** is true, lists active assign codes.

LOCAL [*device specifier / loop*]

Statement: Sets an individual HP-IL device or all HP-IL devices to Local mode. (Controller.)

LOCAL

Sets all HP-IL devices to Local mode (and makes them not Remote enabled).

IF NOT R THEN LOCAL "HP82164A"

If **R** is not true, sets HP-IL/RS-232 interface to Local mode.

LOCAL LOCKOUT

Statement: Sets all HP-IL devices to Local Lockout condition, but only if the system is Remote enabled. (Controller.)

REMOTE @ LOCAL LOCKOUT

Makes the system Remote enabled, then sets all devices to Local Lockout condition.

IF I=10 THEN LOCAL LOCKOUT

If **I** = 10 and if system is Remote enabled, sets all devices to Local Lockout condition.

O

OFF INTR

Statement: Cancels HP-IL interrupt branching. (Controller and device.)

OFF INTR

Disables HP-IL interrupt branching.

IF S THEN OFF INTR

If **S** is true, disables HP-IL interrupt branching.

OFF IO

Statement: Suspends HP-71 I/O operation. (Controller and device.)

OFF IO

Suspends I/O operation.

IF BIT(STATUS,5) THEN OFF IO

If HP-71 isn't controller, then suspends I/O operation.

ON INTR GOTO / GOSUB *line reference*

Statement: Defines how a program branches when an enabled HP-IL interrupt event occurs. (Controller and device.)

ON INTR GOTO 5000

Defines direct branching to line 5000 when an enabled HP-IL interrupt event occurs.

ON INTR GOSUB ISUB

Defines subroutine branching to label **ISUB** when an enabled HP-IL interrupt condition occurs.

OUTPUT *device specifier / loop* [**USING** *image reference*] [**;** *output list*]

Statement: Sends data from numeric and string expressions to HP-IL. (Controller and device.)

OUTPUT 2; "MOVE";N

Sends to device at address 2 a sequence of characters representing the string "MOVE" and the number stored in **N**.

OUTPUT INTRFCE(2); A1,A2

Sends to second interface device a sequence of characters representing the numbers stored in **A1** and **A2**.

OUTPUT "HP82164A" USING "80A"; X\$,Y\$

Sends to the first HP-IL/RS-232 interface (device ID "HP82164A") two 80-character sequences of characters representing the strings stored in **X\$** and **Y\$**.

OUTPUT 3 USING "#,B"; I

Sends to device at address 3 one byte having the value **I**.

OUTPUT LOOP; B1\$

Sends to HP-IL (device not specified) a sequence of characters representing the string stored in **B1\$**.

Free-field output sends characters according to the current round-off setting for numeric values and sends all string characters for string values. A semicolon after an

output item specifies close spacing; a comma specifies wide spacing. An end-of-line sequence is sent after the last item (unless suppressed by a trailing semicolon or comma).

Formatted output sends characters according to specified image symbols (refer to "Reference Tables").

P

PACK *device specifier*

Statement: Packs directory and storage space on a medium. (Controller.)

PACK %16

Packs the medium in the first HP 82161A Digital Cassette Drive (accessory ID 16).

IF V THEN PACK TAPE(2)

If **V** is true, packs the medium in the second mass storage device.

PACKDIR *device specifier*

Statement: Packs only directory space on a medium. (Controller.)

PACKDIR :MASSMEM(3)

Packs the directory of the medium in the third mass storage device.

IF M=10 THEN PACKDIR 1

If **M** = 10, packs the directory of the medium in the device at address 1.

PASS CONTROL [*device specifier / loop*]

Statement: Gives control of the HP-IL system to another device, causing the HP-71 to operate as an HP-IL device (*not* as the controller). (Controller.)

PASS CONTROL :HP71

Gives control of the loop to the first HP-71 (device ID "HP71") after the current HP-71 controller.

PASS CONTROL LOOP

Gives control of the loop to whatever device is a talker.
(The device isn't specified by this statement.)

PRINTER IS *device specifier / loop*

Statement: Assigns one HP-IL device to be used for all printing operations. (Controller and device.)

PRINTER IS PRINTER(2)

Assigns the second printer device to be the **PRINTER IS** device.

PRINTER IS *

Makes the **PRINTER IS** device be the same as the **DISPLAY IS** device.

PRINTER IS NULL

Assigns no **PRINTER IS** device.

PRIVATE *file specifier*

Statement: Permanently prevents a file from being changed or inspected. (Controller.)

PRIVATE PRGMA:2

Makes private the file **PRGMA** in the device at address 2.

PRIVATE "FILE5.EXP1"

Makes private the file **FILE5** on the medium with volume label **EXP1**.

PURGE *file specifier*

Statement: Deletes a file (if it's not secure). (Controller.)

PURGE BACKUP:MASSMEM(2)

Purges the file **BACKUP** in the second mass storage device.

IF F\$=A\$ THEN PURGE A\$

If **F\$** equals **A\$**, purges the file specified by **A\$**.

R

READDDC

Function: Returns the number of the last HP-IL device-dependent command message received. Returns the value -1 if no message received since last checked. (Controller and device.)

X=READDDC

Sets **X** to the number of the last device-dependent message.

IF BIT(READINTR,0) THEN A=READDDC

If bit 0 of the interrupt-cause byte is true (meaning a device-dependent message was received), sets **A** to the number of the last device-dependent message.

Device Dependent Talker messages, numbered from 0 through 31, correspond to **READDDC** values 0 through 31. Device Dependent Listener messages, numbered from 0 through 31, correspond to **READDDC** values 32 through 63.

READINTR

Function: Returns the value of the interrupt-cause byte, which indicates HP-IL interrupt events that have occurred, and then clears that byte. (Controller and device.)

IF BIT(READINTR,4) THEN GOTO 100

If bit 4 of the interrupt-cause byte is true (indicating service request received), branches to line 100.

Interrupt-Cause Byte

Bit	Decimal Value	Interrupt Event	Controller	Device
7	128	Interface Clear	X	X
6	64	Listener	(X)	X
5	32	Controller		X
4	16	Talker	(X)	X
3	8	Service Request	X	
2	4	Device Clear	X	X
1	2	Trigger	X	X
0	1	Device Dependent	X	X

REMOTE [*device specifier / loop*]

Statement: Enables all HP-IL devices to change to Remote mode and can also set an individual HP-IL device to Remote mode. (Controller.)

REMOTE

Enables all HP-IL devices to change to Remote mode.

IF R THEN REMOTE "HP82164A"

If **R** is true, sets HP-IL/RS-232 interface (device ID "HP82164A") to Remote mode.

RENAME *file specifier* **TO** *file specifier*

Statement: Changes the name of a file. (Controller.)

RENAME "FILE08:TAPE" TO "MARGIN"

Changes the name of file **FILE08** in first mass storage device to **MARGIN**.

RENAME "POINTS" TO N\$&":TAPE(2)"

Changes the name of file **POINTS** in second mass storage device to name specified by **N\$**.

REQUEST *status byte*

Statement: Defines the HP-71 status byte that is sent when serially polled by an HP-IL controller (while operating as a device). (Device.)

REQUEST 224

Defines the value 224 (a system status byte indicating "request control of loop") to be sent as the HP-71 status byte when serially polled. Also requests service from controller.

IF FLAG(−61) THEN REQUEST 129

If flag −61 (BAT annunciator on) is true, defines the value 129 (a system status byte indicating "low battery") to be sent as the HP-71 status byte when serially polled.

(Refer to "Reference Tables" for definitions of system status bytes.)

RESET HPIL

Statement: Resets the HP-IL interface to a known condition. (Controller and device.)

RESET HPIL

Resets the HP-IL interface.

IF LEN(A\$)>L THEN RESET HPIL

If the length of **A\$** is greater than **L**, resets the HP-IL interface.

RESTORE IO

Statement: Enables I/O operations to occur on HP-IL. (Controller and device.)

RESTORE IO

Enables I/O operations on HP-IL.

IF BIT(STATUS,5) THEN RESTORE IO

If bit 5 of the HP-IL status byte is true (the HP-71 is controller), sets conditions for I/O operations.

RUN *file specifier*

Statement: Copies specified file into main RAM and begins executing that file. (Controller and device.)

RUN FAST:TAPE(2)

Loads and runs program **FAST** from second mass storage device.

IF A THEN RUN PT3.NOM

If **A** is true, loads and runs program **PT3** from medium with volume label **NOM**.

S

SECURE *file specifier*

Statement: Prevents a file from being altered or purged. (Controller.)

SECURE "RATES:MASSMEM(2)"

Secures the file **RATES** in the second mass storage device.

SECURE ROCKIN.ROLL

Secures the file **ROCKIN** on the medium with volume label **ROLL**.

SEND *message list*

Statement: Sends individual HP-IL messages on the loop. (Controller and device.)

SEND IFC

Sends an Interface Clear message, clearing all talkers and listeners.

SEND UNT UNL LISTEN 2,6,7 MTA

Sends message sequence that sets up listeners at addresses 2, 6, and 7, and makes the HP-71 a talker. The HP-71 can now send data to three listeners using **OUTPUT LOOP**.

SEND UNT UNL CMD 146 LISTEN A1 MTA DATA "P4;" UNT UNL CMD 147

Makes the system Remote enabled (command 146), makes device at address **A1** a listener (also setting it to Remote mode), sends it the Remote instruction "P4;" (meaning "no parity" to HP-IL/RS-232 interface), then makes system not Remote enabled (command 147).

SEND UNL CMD C\$ LISTEN 1 SAD 2,4 MTA DATA X\$ UNT UNL

Sends HP-IL command message(s) specified by **C\$**, makes devices at extended addresses 1.03 and 1.05 listeners, makes the HP-71 a talker, and sends data specified by **X\$**.

The following message indicators and qualifiers specify the messages to be sent. Refer to the owner's manual for precautions regarding the use of **SEND** while the HP-71 is a controller or a device.

Message Indicators and Qualifiers

Indicator	Qualifier	Message
Command Group:		
CMD	byte number	Command message.
	byte string	Sequence of command messages.
DDL	byte number	Device-Dependent Listener message.
DDT	byte number	Device-Dependent Talker message.
GTL		Go To Local message.
IFC		Interface Clear message.
LISTEN	byte number	Listen Address message.
LPD		Loop Power Down message.
MLA		No message sent (HP-71 becomes listener).
MTA		Untalk message (HP-71 becomes talker).
SAD	byte number	Secondary Address message.
SDC		Selected Device Clear message.
TALK	byte number	Talk Address message.
UNL		Unlisten message.
UNT		Untalk message.
Ready Group:		
RDY	byte number	Ready message.
Identify Group:		
IDY	byte number	Identify message.
Data/End Group:		
DATA	byte number	Data Byte message.
	byte string	Sequence of Data Byte messages.
END	byte number	End Byte message.
EOL		Sequence of Data Byte messages defined by the ENDLINE statement. (May be used only immediately after a DATA byte number or string.)

SPOLL(*device specifier* **)**

Function: Returns a value that represents one or more status bytes from an HP-IL device. Returns the value -1 if there's no response. (Controller.)

A=SPOLL(I)

Sets **A** to the value representing the status of the device at address **I**.

IF SPOLL("TAPE(2)")=20 THEN 100

If the status of the second mass storage device equals 20 (no tape installed in cassette drive), then branches to line 100.

(Refer to "Reference Tables" for definitions of system status bytes.)

STANDBY ON / OFF / *timeout period* [, *verify interval*]

Statement: Sets the HP-IL timeout period and verify interval. (Controller and device.)

STANDBY 15

Sets the timeout period and verify interval to 15 seconds.

IF S THEN STANDBY ON

If **S** is true, sets the timeout period and verify interval to infinity (HP-71 waits forever for HP-IL response).

STANDBY OFF

Sets timeout period to 60 seconds and verify interval to 2 seconds (default).

STATUS

Function: Returns the HP-IL interface status. (Controller and device.)

S=STATUS

Sets **S** to the value of the HP-IL interface status.

IF BIT(STATUS,5) THEN 200

If bit 5 of the HP-IL interface status is true (HP-71 is controller), branches to line 200.

HP-IL Interface Status Byte

Bit	Decimal Value	Condition	Controller	Device
7	128	Data Transfer	X	
6	64	Listener	X	X
5	32	Controller	X	
4	16	Talker	X	X
3	8	Service Request	X	
2	4	Asynchronous Requests	X	X
1	2	Remote	(X)	X
0	1	Local Lockout	X	X

T

TRANSFORM [*file specifier*] **INTO BASIC/TEXT** *file specifier*

Statement: Creates new file of specified type from source file. (Controller.)

TRANSFORM PG1:TAPE INTO TEXT PRG

Creates TEXT file **PRG** in main RAM from file **PG1** in first mass storage device.

TRANSFORM TVER.LIST INTO BASIC BVER.LIST

Creates BASIC file **BVER** from file **TVER**, both on medium with volume label **LIST**.

TRIGGER [*device specifier / loop*]

Statement: Triggers an event at an HP-IL device. (Controller.)

IF T THEN TRIGGER 3

If **T** is true, triggers the device at address 3.

TRIGGER LOOP

Triggers all devices that are set to listener status.

UNSECURE *file specifier*

Statement: Cancels security for a file, allowing it to be altered or purged. (Controller.)

UNSECURE PARKER.ROLL

Cancels security for the file **PARKER** on the medium with volume label **ROLL**.

Reference Tables

System status bytes are status byte values having meanings that are defined according to standardized HP-IL conventions.

System Status Bytes

Status Byte Value		
Service Not Requested	Service Requested	Meaning
Events:		
134	198	Self-test failure.
136	200	Powering down.
137	201	External service request.
130	194	Manual intervention required.
131	195	Data error.
135	199	Command error.
133	197	No room for data.
132	196	Device error.
129	193	Low battery.
138	202	Device-dependent service request.
159	223	ASCII display follows.
States:		
160	224	Request control of loop.
162	226	Ready to send data.
161	225	Ready to receive data.
163	227	Not ready to receive or send data.
128	192	All okay.

Formatted I/O operations use image symbols to define the way that characters are to be interpreted as they're sent using **OUTPUT** or received using **ENTER**.

Image Symbols for OUTPUT and ENTER

Image Symbol	OUTPUT Operation	ENTER Operation	May Be Replicated
Numeric I/O:			
D	Digit	Digit	Yes
Z	Digit	Digit	Yes
*	Digit	Digit	Yes
.	Decimal	Digit	No
S	Sign	Digit	No
M	Sign	Digit	No
E	Exponent	Five Digits	No
C	Comma	Comma	No
R	Comma Radix*	Comma Radix*	No
P	Period Separator*	Period Separator*	No
K	Compact	Free-Field	No
H	Compact*	Free-Field*	No
^	Hide	Hide	No
String I/O:			
A	Character	Character	Yes
K	Compact	Free-Field	No
H	Compact	Free-Field	No
^	Hide	Hide	No
Binary I/O:			
B	Byte	Byte	No
Special Symbols:			
X	Blank	Ignore	Yes
"..."	Quotes	Ignore	Yes
'...'	Quotes	Ignore	Yes
,	Field Separator	Field Separator	No
@	Form Feed	Field Separator	Yes
/	Line Ending	Line Feed	Yes
(...)	Group	Group	Yes
#	Terminator	Terminator	No
* For European formatting.			

The table below lists ASCII characters and HP-71 characters for character codes 0 through 127. For character codes 128 through 255, HP-71 characters are identical to those for 0 through 127 except for five marked by footnotes. (ASCII characters are defined only for codes 0 through 127.) Binary representations for codes 128 through 255 have the first bit equal to “1”.

Character Sets

Character Code	Binary	ASCII Character	HP-71 Character	HP-71 Keystrokes
0	00000000	(NUL)	(none) ¹	[g][CTRL][g][@]
1	00000001	(SOH)	▪	[g][CTRL][A]
2	00000010	(STX)	⌘	[g][CTRL][B]
3	00000011	(ETX)	↵	[g][CTRL][C]
4	00000100	(EOT)	␣	[g][CTRL][D]
5	00000101	(ENQ)	␣	[g][CTRL][E]
6	00000110	(ACK)	␣	[g][CTRL][F]
7	00000111	(BEL)	␣	[g][CTRL][G]
8	00001000	(BS)	(none) ²	[g][CTRL][H]
9	00001001	(HT)	␣	[g][CTRL][I]
10	00001010	(LF)	(none) ³	[g][CTRL][J]
11	00001011	(VT)	␣	[g][CTRL][K]
12	00001100	(FF)	␣	[g][CTRL][L]
13	00001101	(CR)	(none) ⁴	[g][CTRL][M]
14	00001110	(SO)	␣	[g][CTRL][N]
15	00001111	(SI)	␣	[g][CTRL][O]
16	00010000	(DLE)	␣	[g][CTRL][P]
17	00010001	(DC1)	␣	[g][CTRL][Q]
18	00010010	(DC2)	␣	[g][CTRL][R]
19	00010011	(DC3)	␣	[g][CTRL][S]
20	00010100	(DC4)	␣	[g][CTRL][T]
21	00010101	(NAK)	␣	[g][CTRL][U]
22	00010110	(SYN)	␣	[g][CTRL][V]
23	00010111	(ETB)	␣	[g][CTRL][W]

Character Code	Binary	ASCII Character	HP-71 Character	HP-71 Keystrokes
24	00011000	(CAN)	ö	[9] [CTRL] [X]
25	00011001	(EM)	ü	[9] [CTRL] [Y]
26	00011010	(SUB)	û	[9] [CTRL] [Z]
27	00011011	(ESC)	(none) ⁵	[9] [CTRL] [9] [I]
28	00011100	(FS)	Σ	(none)
29	00011101	(GS)	≠	[9] [CTRL] [9] [J]
30	00011110	(RS)	£	[9] [CTRL] [9] [^]
31	00011111	(US)	⌘	(none)
32	00100000	(space)	(space)	[SPC]
33	00100001	!	!	[9] [!]
34	00100010	"	"	[9] ["]
35	00100011	#	#	[9] [#]
36	00100100	\$	\$	[9] [\$]
37	00100101	%	%	[9] [%]
38	00100110	&	&	[9] [&]
39	00100111	,	,	[9] [']
40	00101000	(([9] [(
41	00101001))	[9] [)]
42	00101010	*	*	[9] [*]
43	00101011	+	+	[9] [+]
44	00101100	,	,	[9] [,]
45	00101101	—	—	[9] [-]
46	00101110	.	.	[9] [.]
47	00101111	/	/	[9] [/]
48	00110000	0	0	[9] [0]
49	00110001	1	1	[9] [1]
50	00110010	2	2	[9] [2]
51	00110011	3	3	[9] [3]
52	00110100	4	4	[9] [4]
53	00110101	5	5	[9] [5]
54	00110110	6	6	[9] [6]

Character Code	Binary	ASCII Character	HP-71 Character	HP-71 Keystrokes
55	00110111	7	7	
56	00111000	8	8	
57	00111001	9	9	
58	00111010	:	:	
59	00111011	;	;	
60	00111100	<	<	
61	00111101	=	=	
62	00111110	>	>	
63	00111111	?	?	
64	01000000	@	@	
65	01000001	A	A	
66	01000010	B	B	
67	01000011	C	C	
68	01000100	D	D	
69	01000101	E	E	
70	01000110	F	F	
71	01000111	G	G	
72	01001000	H	H	
73	01001001	I	I	
74	01001010	J	J	
75	01001011	K	K	
76	01001100	L	L	
77	01001101	M	M	
78	01001110	N	N	
79	01001111	O	O	
80	01010000	P	P	
81	01010001	Q	Q	
82	01010010	R	R	
83	01010011	S	S	
84	01010100	T	T	
85	01010101	U	U	

Character Code	Binary	ASCII Character	HP-71 Character	HP-71 Keystrokes
86	01010110	V	V	
87	01010111	W	W	
88	01011000	X	X	
89	01011001	Y	Y	
90	01011010	Z	Z	
91	01011011	[[
92	01011100	\	\	(none)
93	01011101]]	
94	01011110	^	^	
95	01011111	_	_	(none)
96	01100000	`	`	(none)
97	01100001	a	a	
98	01100010	b	b	
99	01100011	c	c	
100	01100100	d	d	
101	01100101	e	e	
102	01100110	f	f	
103	01100111	g	g	
104	01101000	h	h	
105	01101001	i	i	
106	01101010	j	j	
107	01101011	k	k	
108	01101100	l	l	
109	01101101	m	m	
110	01101110	n	n	
111	01101111	o	o	
112	01110000	p	p	
113	01110001	q	q	
114	01110010	r	r	
115	01110011	s	s	
116	01110100	t	t	

Character Code	Binary	ASCII Character	HP-71 Character	HP-71 Keystrokes
117	01110101	u	␣	[g][u]
118	01110110	v	␣	[g][v]
119	01110111	w	␣	[g][w]
120	01111000	x	␣	[g][x]
121	01111001	y	␣	[g][y]
122	01111010	z	␣	[g][z]
123	01111011	{	␣	[g][{]
124	01111100			(none)
125	01111101	}	␣	[g][}]
126	01111110	~	~	(none)
127	01111111	(DEL)	␣	(none)

¹ Character 128 is (space).
² Character 8 performs a display function; character 136 is E_2 .
³ Character 10 performs a display function; character 138 is E_4 .
⁴ Character 13 performs a display function; character 141 is E_1 .
⁵ Character 27 performs a display function; character 155 is E_3 .

Flag Summary

The HP-IL interface uses four HP-71 system flags. Their effects are summarized below.

- | Flag | Purpose |
|--------------------------------|--|
| —21 Inhibit Power Down | <p>Clear: Turns off devices when HP-71 turned off, considers addresses invalid.</p> <p>Set: Doesn't affect devices, considers addresses still valid.</p> |
| —22 Extended Addressing | <p>Clear: HP-71 uses simple addressing.</p> <p>Set: HP-71 uses extended addressing first, simple addressing second.</p> |

—23 End Of Transmission

Clear: End Of Transmission doesn't affect **ENTER**.

Set: End Of Transmission terminates variable assignment for **ENTER**.

—24 Inhibit Addressing

Clear: HP-71 assigns new addresses if old addresses invalid.

Set: HP-71 doesn't automatically assign new addresses.

Error Summary

Error Number	Message and Condition
255001	ASSIGN IO Needed No devices assigned.
255003	Excess Chars Excess characters in statement.
255004	Missing Parm Parameter missing in statement or function.
255005	Invalid Parm Parameter is improper in statement or function.
255006	Invalid Expr Invalid expression prevents evaluation.
255007	Syntax Improper syntax.
255016	File Protect Improper operation on secure or private file.
255017	End of Medium File too big to store or medium full.
255018	Invalid Medium Mass storage drive motor stalled.
255019	Invalid Medium Medium not initialized to proper format.
255020	No Medium No medium installed in drive.

- 255022 **File Not Found**
 Specified file not found in device.
- 255023 **Invalid Medium**
 Mass storage device opened during operation.
- 255024 **Invalid Medium**
 Medium not initialized.
- 255025 **Invalid Medium**
 Directory record number error.
- 255026 **Invalid Medium**
 Data checksum error on medium.
- 255028 **Size of File**
 File too large to copy.
- 255030 **File Exists**
 File with specified name already exists.
- 255031 **Directory Full**
 Directory space is full.
- 255032 **Device Not Found**
 Specified device not found in loop.
- 255034 **Device Not Ready**
 Device didn't respond.
- 255035 **Loop Broken**
 Loop continuity broken.
- 255036 **Message Error**
 Too many messages received.
- 255037 **Message Error**
 Message lost due to slow retransmission.
- 255038 **Message Error**
 Message altered during transmission.
- 255039 **Unexpected Message**
 Unexpected message sent by device.
- 255040 **Message Error**
 Too many messages received.
- 255041 **Invalid Mode**
 HP-71 not controller or device, as required.
- 255042 **Loop Broken**
 Message transmission interrupted.

- 255043 **Loop Broken**
Transmission exceeded **STANDBY** timeout period.
- 255044 **System Error**
Invalid address (flag —24 set); internal I/O channel error.
- 255045 **Self Test Failed**
Interface failed its self-test.
- 255047 **Device Type**
Improper type of device for operation.
- 255052 **Aborted**
Manually interrupted operation.
- 255053 **Invalid Device Spec**
Improper device specifier used.
- 255054 **Data Type**
Improper data type specified.
- 255056 **Invalid Arg**
Argument out of range.
- 255057 **No Loop**
Interface isn't installed.
- 255059 **Insufficient Memory**
Not enough memory available for operation.
- 255060 **RESTORE IO Needed**
I/O operation not restored.

Keyword Index

System Setup:

ASSIGN IO	2	RESET HPIL	18
LIST IO	11	RESTORE IO	18
OFF IO	12		

Printer and Display:

DISPLAY IS	8	PRINTER IS	15
------------	---	------------	----

Mass Storage:

ASSIGN #	2	PACKDIR	14
CAT	4	PRIVATE	15
CAT\$	5	PURGE	15
CHAIN	5	RENAME	17
COPY	6	RUN	18
CREATE	6	SECURE	18
INITIALIZE	11	TRANSFORM	22
PACK	14	UNSECURE	23

General I/O—Data Transfer:

ENTER	10	OUTPUT	13
-------	----	--------	----

General I/O—HP-IL Interaction:

CLEAR	5	READDDC	16
DEVADDR	7	READINTR	16
DEVAID	8	REMOTE	17
DEVID\$	8	REQUEST	17
ENABLE INTR	9	SEND	19
LOCAL	11	SOLL	21
LOCAL LOCKOUT	12	STANDBY	21
OFF INTR	12	STATUS	21
ON INTR	13	TRIGGER	22

General I/O—Passing Control:

CONTROL OFF/ON	5	PASS CONTROL	14
----------------	---	--------------	----

Binary Functions:

BINAND	3	BINIOR	4
BINCMP	3	BIT	4
BINEOR	3		



Portable Computer Division

1000 N.E. Circle Blvd., Corvallis, OR 97330, U.S.A.

© Hewlett-Packard Company 1983

82401-90010 Rev. B English

Printed in Singapore 1/84