

3. ELECTRICAL SPECIFICATIONS

3.1 General

This chapter presents the detailed electrical specifications for HP-IL. As discussed in chapter 1, the link between devices on the loop is a two-wire electrically balanced line. One of the conductors is designated the reference line and all voltage measurements are made with respect to the reference.

A device output must drive only one device input. This allows certain advantages over a bus type structure. Namely, the input may change the electrical signal to its own convenience without affecting any other device. Therefore, the specification thoroughly specifies a device output while putting only minimal constraints on a device input. Test procedures are specified for verifying the proper operation of input and output circuits to insure system compatibility.

Additionally, there may be instances where the designer requires certain additional capabilities from the link between devices that are not provided within this specification, such as longer distance or better noise immunity. As long as the waveforms and impedances at an HP-IL mechanically compatible output or input terminal are in compliance with the specification, the designer may choose any implementation desired. This might include level translation, data rate conversion, transmission media conversion, etc. For example, longer distances between devices could be achieved by permanently or uniquely attaching a cable to a special device output as long as the signal at the HP-IL compatible connectors are in compliance with the output specifications of section 3.3.

Any link with mechanically compatible HP-IL connectors, cables, etc. must also be functionally and electrically compatible with these specifications to prevent inadvertent connection to nonstandard links. Externally inaccessible links, as between multiple devices within a single package, may be implemented in any manner deemed appropriate.

These specifications are in general based upon an implementation using simple pulse transformers within the device. One transformer provides isolation and level conversion from the input terminals to the HP-IL receiver electronics and a second transformer performs a similar operation for the output terminals. While this specification is not intended to preclude implementations which do not use transformers, it is strongly recommended that transformers be used. It may in fact be quite difficult or even impossible to achieve these specifications with other technology.

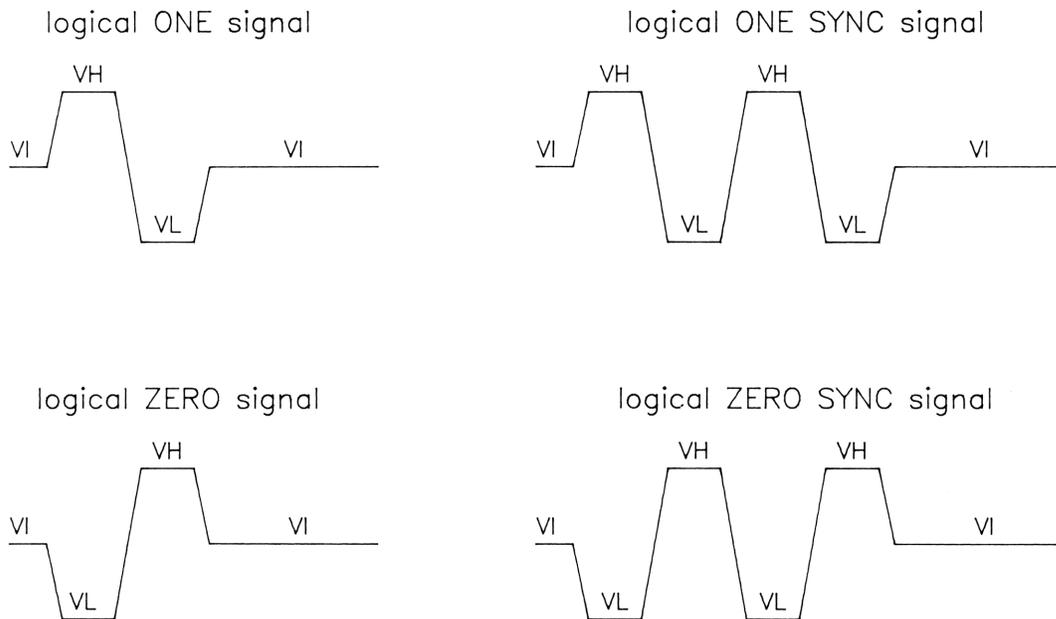


Figure 3-1. HP-IL Electrical and Logical Relationships

3.2 Electrical and Logical Relationships

There are four distinct logical states defined by HP-IL as shown in figure 3-1. VH, VL, and VI are the voltage levels of the high, low, and idle times respectively. The detailed level and timing specifications are given in section 3.3.

The relationship between remote messages and the logical states of the interface is as follows: The frame is divided into a 'SYNC' code followed by ten (10) logical codes, each of which is a '0' or '1'. Encoded in the 'SYNC' code is the first bit of the frame, C2, followed by C1, C0, D7, D6, D5, D4, D3, D2, D1, and D0. A sample frame is shown in figure 3-2.

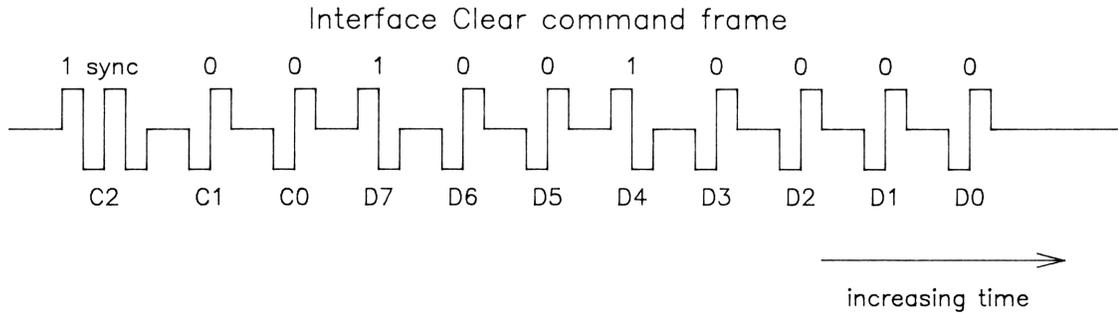


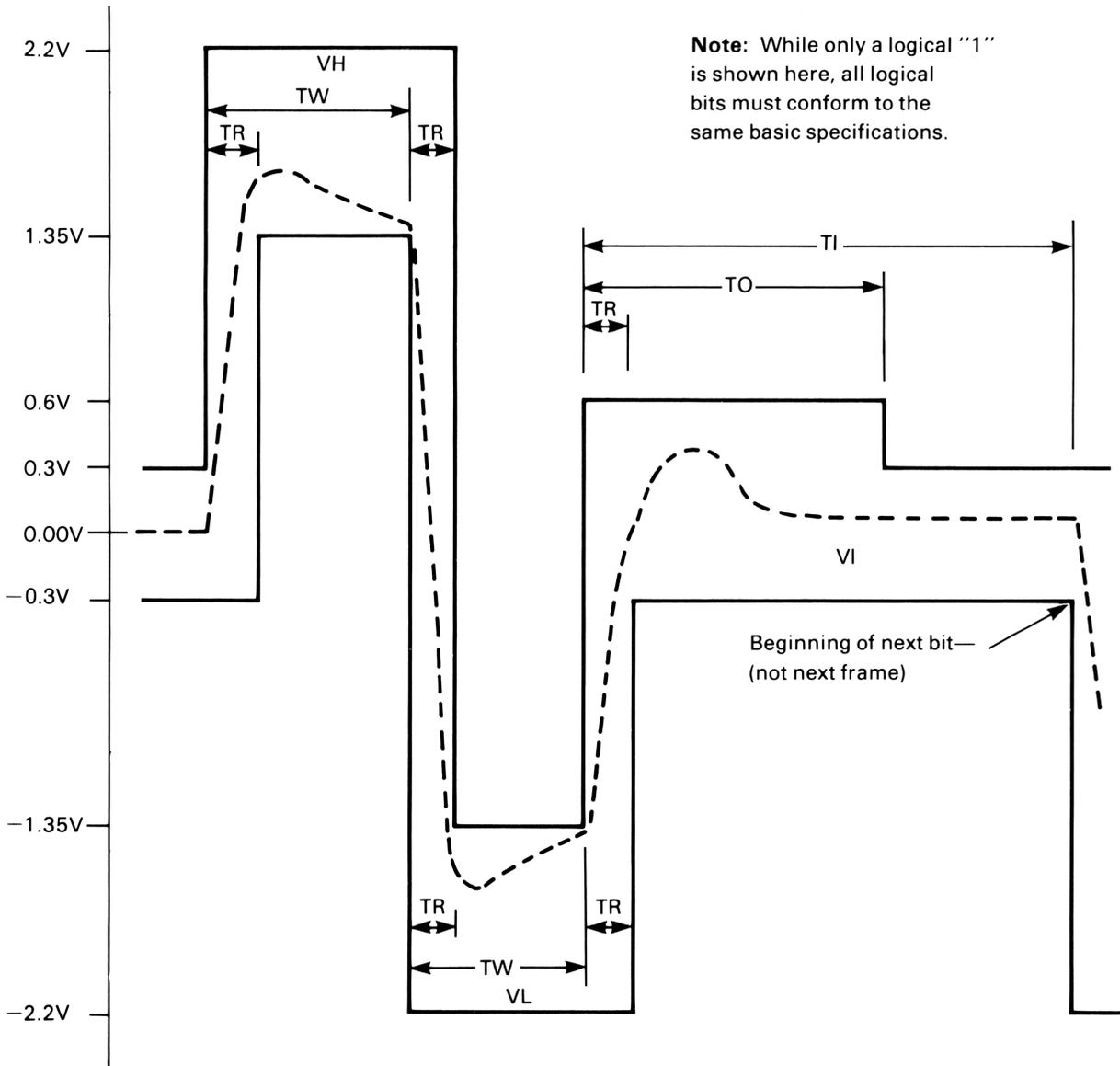
Figure 3-2. Sample HP-IL Frame

3.3 Output Specifications

The output is intended to drive a 100 ohm balanced transmission line of any distance up to 100 meters in length. The design is based on a matched source and unmatched load so that transmission line reflections are absorbed at the source (output terminals). The output is thoroughly specified in the form of a Thevenin equivalent circuit so that the input circuit design is given maximum information and freedom.

3.3.1 Open Circuit Output Voltage

The open circuit output voltage waveform shall remain within the boundaries specified in figure 3-3. The measurement shall be made at the device output terminals or at the end of an HP-IL cable no more than one meter in length. If the HP-IL cable is permanently or uniquely attached to the device, the measurement shall be made at the cable output, even if its length is greater than one meter (this allows for special drivers to drive very long cables). It is recommended that the measurement be made with a balanced differential oscilloscope such that the impedance from each output terminal to earth ground is equal.



Parameter	Minimum	Maximum
TR	30nS	120nS (200 nS*)
TI	1.9 μ S	10 μ S
TO		1 μ S
TW	950nS	1500nS

*When tested with a 20 meter cable.

Figure 3-3. Open Circuit Output Waveform

Note: A device which only barely meets the waveform window of figure 3-3 is quite likely to fail a waveform test at 20 meters unless the output impedance can perfectly absorb all transmission line reflections. This would require output impedance to be identical to the characteristic impedance of the cable. Refer to section 3.3.2 for the test procedure of reflection effects with a 20 meter cable.

The pulse width, TW, shall not vary from its mean value by more than 10 percent within a single frame.

While the idle voltage, VI, extends to ± 0.3 volts, the DC voltage level shall approach 0 volts to prevent saturation of an input circuit transformer. The large window for VI is intended to allow for transient distortion caused by ringing, reflections, etc. in the output circuits.

The idle voltage, VI, shall remain valid for at least 5 microseconds between frames.

3.3.2 Output Impedance

The output impedance shall remain within the boundaries of magnitude and phase specified in figure 3-4 when the HP-IL voltage levels are at the VH, VL, or VI levels. The measurement shall be made between the two lines of the output, or at the end of an HP-IL extension cable no more than one meter in length. Conventional AC impedance meters are not capable of measuring impedance during the VH and VL pulse times due to the narrow pulse width and variable test signal amplitude. Therefore, it is required that the output meet the tests outlined throughout the rest of this section to insure system compatibility. It is also the purpose of these tests to allow the maximum amount of design flexibility by tolerating a significant amount of parasitic reactance and/or non-linearity so that system size and cost may be kept to a minimum. However, the allowance for design flexibility requires more thorough testing than a more restrictive specification.

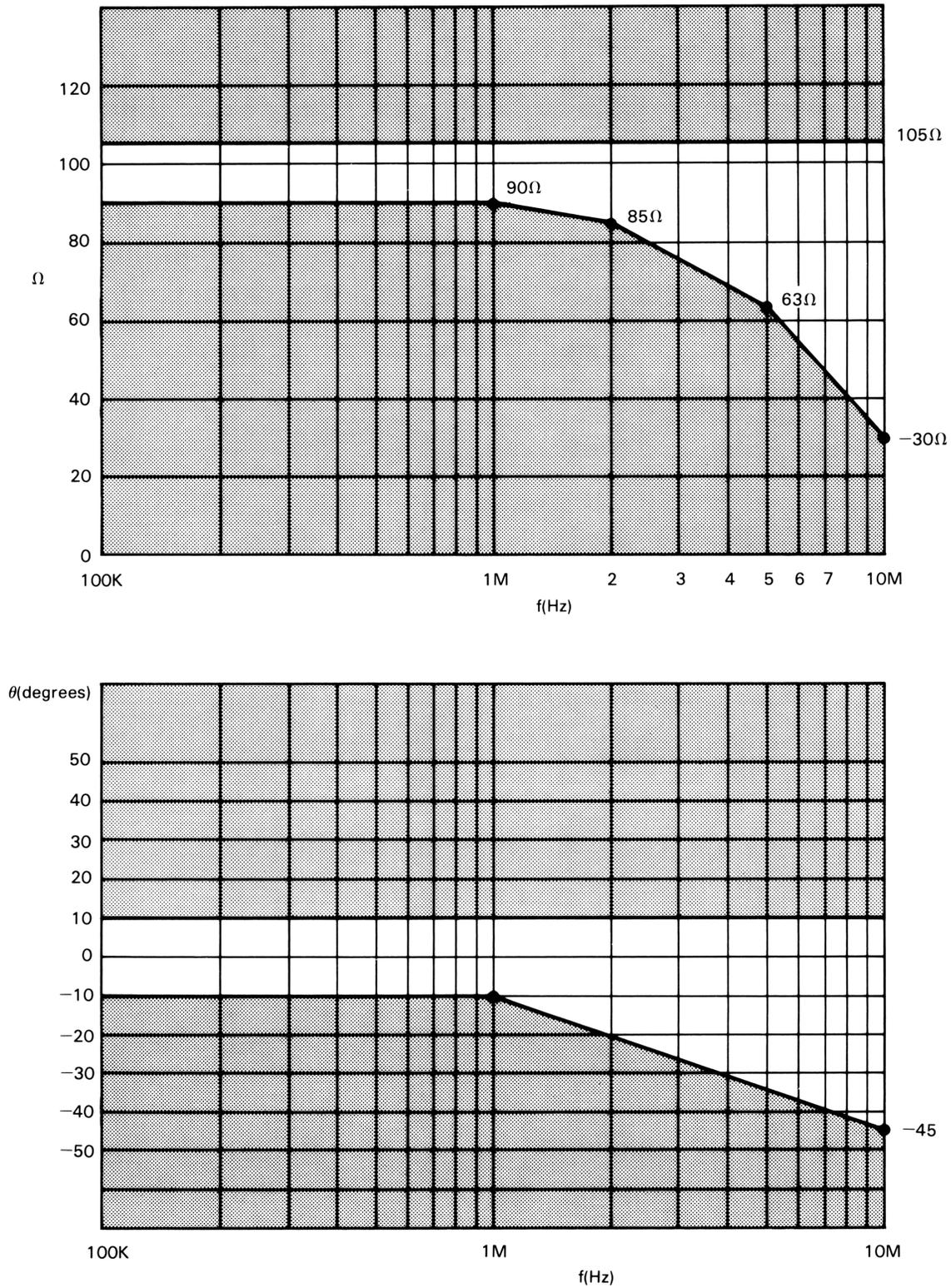


Figure 3-4. Output Impedance Magnitude and Phase

Output Impedance - Pulse Attenuation and Droop Effects

At frequencies under 1 MHz, the output impedance is intended to put minimum and maximum boundaries on pulse attenuation caused by resistive loading at the input terminals of an HP-IL device, and also to put a limit on pulse droop caused by shunt inductive loading (due to transformer) at the input terminals. The actual amount of attenuation or droop is dependent on the input designer's choice of input resistance and inductance. Figure 3-5 outlines the test procedure to verify compliance. The test is done without an HP-IL input connected.

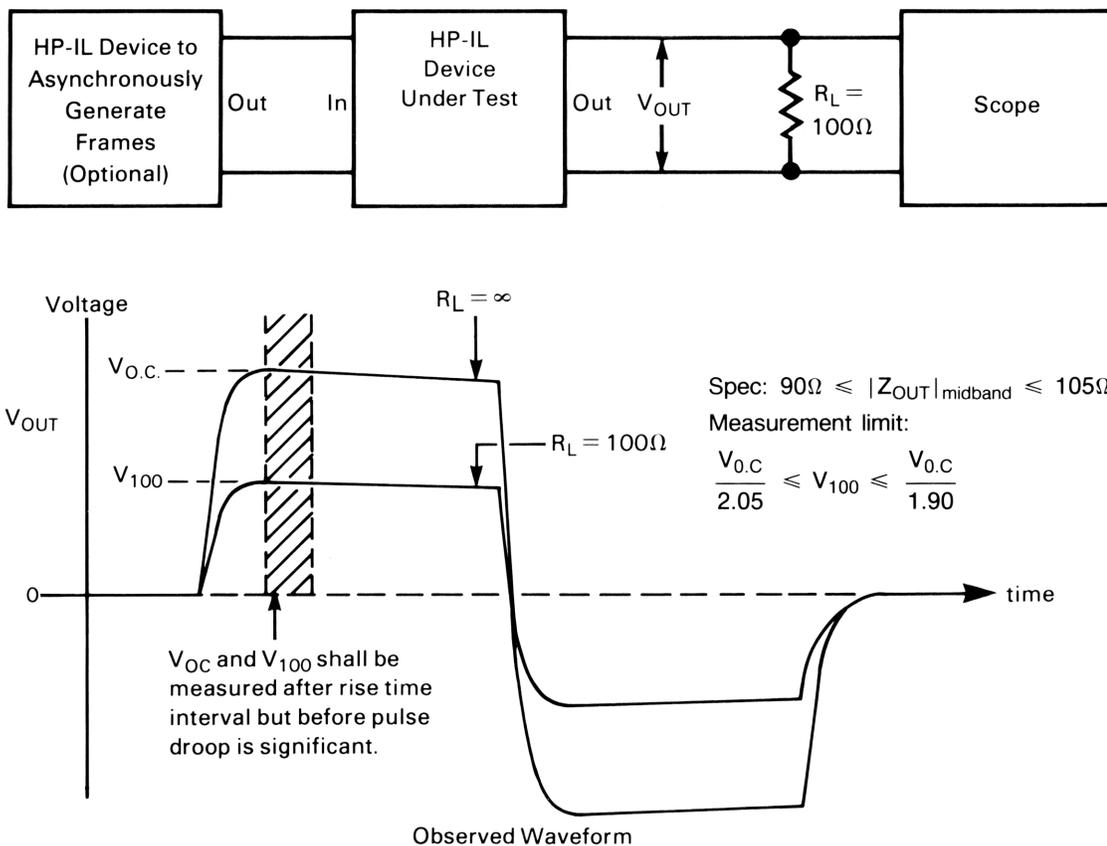


Figure 3-5. Pulse Attenuation and Droop Test Procedure

The device under test must generate a frame(s) which allows voltage measurement of V_{out} . For convenient oscilloscope measurement, it is recommended that a single frame be repetitively generated. Since there is no input connected to the output under test, frames must be generated asynchronously by the device under test or by another HP-IL device if the device under test will repeat the frames. This may require some special device software since no handshake will occur.

Output Impedance - Resonance Effects

At short distances between devices, pulse distortion can occur in the form of pulse ringing or overshoot due to the presence of an underdamped RLC type circuit. Because of the probable presence of transformers in both the output and input circuits, the presence of parasitic inductance (L) and capacitance (C) is inevitable. The design objective is to keep the amount of ringing to an acceptable level. The output circuit is specified so that it does not contribute any significant parasitic inductance (by restricting positive phase shift to less than 10 degrees) at frequencies of importance to the input circuit (under 10 MHz). The output circuit is also specified to restrict the amount of parasitic capacitance (which could resonate with an input parasitic inductance) by putting lower limits on impedance magnitude and phase shift in the 1 to 10 MHz range. Above 10 Mhz, the requirement that the waveform risetime be no faster than 30ns should provide adequate bandwidth limiting so that resonance may be ignored.

To test the above criteria, the following procedure is suggested: with the HP-IL output at the VI state (idle state), connect a vector impedance meter (e.g. HP 4275A or HP 4815A or equivalent) to the output terminals and measure magnitude and phase per figure 3-4 at frequencies from 100 KHz to 10 MHz at a test signal amplitude of 0.1 Vac. If it can be demonstrated that the output impedance is independent of test signal amplitude, a different amplitude may be used.

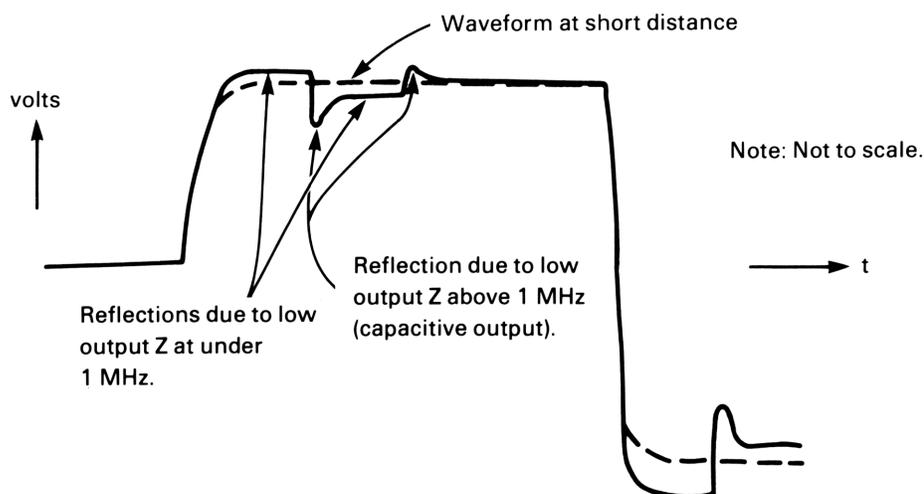


Figure 3-6. Output Waveform With Reflections

Output Impedance - Reflection Effects

At relatively long distances between devices (more than 10 meters), the output impedance is intended to sufficiently match the cable characteristic impedance (100 ohms) so that transmission line reflections are adequately absorbed. For example, an output impedance which barely meets the lower limit of impedance magnitude (figure 3-4) might generate a waveform at the end of a long cable as shown in figure 3-6.

Non-linearities in the output impedance might also cause reflections which a small signal impedance measurement might overlook. Therefore, it is necessary that the output voltage waveform remain within the boundaries of figure 3-3 when tested according to figure 3-7. The device under test must generate frames which allow measurement of the output voltage. Another device may be used for this purpose as shown in figure 3-5. The 20 meter cable is long enough to generate distinct reflections, yet short enough to prevent cable losses from filtering spike type reflections caused by a reactive source impedance. The capacitive load is representative of filtering required of the input circuits of high frequency spike type distortion.

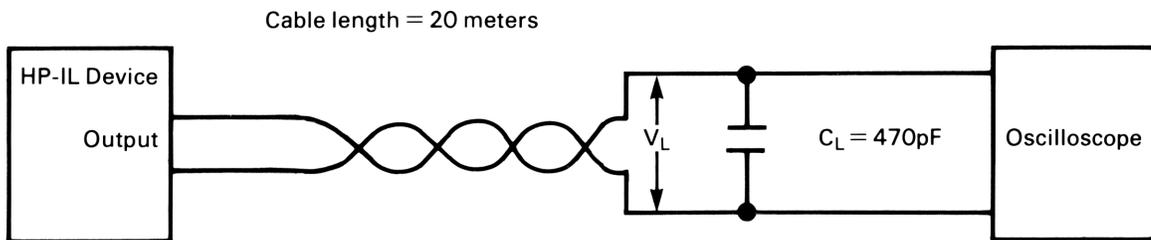


Figure 3-7. Output Impedance Test - Reflection Effects

3.3.3 Common Mode Output Voltage

Since the loop is not perfectly isolated from earth, there is the possibility that during normal operation a device could source common mode signals with respect to earth. The input circuitry is required to reject a certain amount of common mode signal (refer to section 3.7), but above that amount errors can occur. In addition, common mode signals can cause currents to flow through the capacitance present at the loop inputs if the signal is changing with time. Therefore, the amount of common mode signal that the output circuitry can source is restricted to a level that all input circuits can reject. In addition, the rate of change of the common mode signal is limited.

It should be emphasized that these problems can occur during normal operations of the device. There will always be abnormal conditions that could cause a device's output signal to exceed the allowable common mode specifications (e.g. electrostatic discharge).

Under normal operating conditions, the common mode signal sourced by the device under test should meet the restrictions listed in figure 3.8.

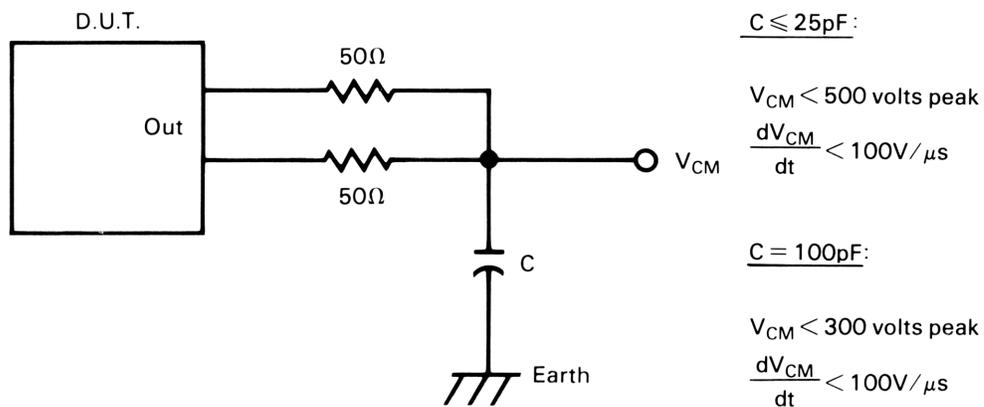


Figure 3-8. Common Mode Output Voltage Test

3.4 Input Specifications

Because each HP-IL output drives only one input, the circuit designer is given as much freedom as possible to detect the specified output waveform using whatever attenuation or waveshaping techniques desired. However, this requires the designer to thoroughly understand how the voltage waveform is modified by different combinations of output, input, and interface cable circuits.

3.4.1 Input Impedance

To limit the amount of power dissipation of the sourcing device, the input resistance shall be at least 100 ohms when measured at 100 KHz at a voltage level of up to 0.85 volts AC (2.4 volts peak to peak). At voltage levels greater than 1.2 volts or less than -1.2 volts, the input impedance is not specified. This allows an input to clamp the voltage, if desired, for waveshaping, electrostatic discharge protection, etc. The chosen level is above the maximum voltage generated by the output circuit if the input is a totally linear 100 ohms.

The input reactance is not specified. However, the input designer must use care to insure that any input reactances, parasitic or otherwise, do not create any waveform distortion which might adversely affect signal detection reliability.

3.4.2 Input Hysteresis

To provide a minimum level of immunity to externally induced noise or internally generated distortion, it is strongly recommended that the input circuits use a schmitt trigger type input with hysteresis.

3.4.3 Input Test Circuits

Because the input circuit thresholds or timing requirements are not specified, it is necessary to provide a means of testing the input circuits for proper operation in all possible HP-IL system configurations. To achieve this, test circuits are specified which are representative of several worst case configurations. While these circuits do not cover all possible worst case configurations, they represent a minimum level of testing to insure compliance.

Table 3-1 lists the characteristics of 5 test configurations to be used to test the input of the device. With the exception of pulse amplitude and output impedance phase, the output parameters are taken directly from the worst case output specifications of section 3.3. A modest amount of noise margin has been included for pulse amplitude and impedance phase to allow for test measurement error and to improve system reliability.

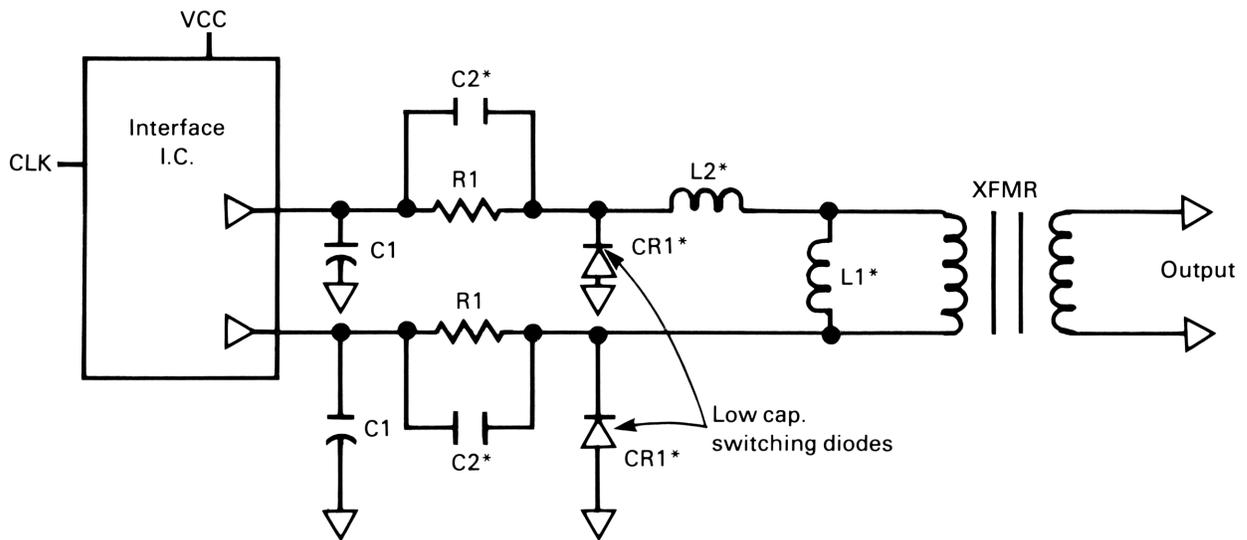
Table 3-1. Input Test Configurations

	Test Circuit	Open Circuit Output Waveform (see figure 3-3)					Output Impedance			Cable Length
		Pulse Amplitude (VH,VL)	Idle Amplitude (VI)	Rise Time (TR)	Pulse Width (TW)	Idle Time (TI)	Magnitude (Z)	Phase (θ)	Test Frequency	
1	1	1.30v	—	120ns	950ns	1900ns	105 ohm	0 °	1 Mhz	0 m
2	1	1.30v	—	120ns	950ns	1900ns	105 ohm	0 °	1 Mhz	100 m
3	2	2.30v	max.*	< 50ns	1500ns	—	90 ohm 64(-3dB)	0 ° -450 °	100 KHz 3 MHz	0 m
4	2	2.30v	max.*	< 50ns	1500ns	—	90 ohm 64(-3dB)	0 ° -450 °	100 KHz 3 MHz	20 m
5	2	2.30v	max.*	< 50ns	1500ns	—	90 ohm 64(-3dB)	0 ° -450 °	100 KHz 3 MHz	100 m

* Recommended only – may be difficult to generate

The two test circuits required to generate the necessary worst case waveforms may be obtained by modifying an HP-IL device's output circuits to meet the requirements of table 3-1. While the means to achieve this are very device dependent, an example is shown in figure 3-9.

The first test circuit is primarily intended to guarantee the detection of the pulse levels VH and VL, the idle level VI, and to guarantee that the VH to VL transition is not detected as a VI state between bits. The second test circuit is primarily intended to guarantee that extra pulse levels, VH and VL, are not detected following a bit. This might cause a normal bit to look like a SYNC bit. This could be caused by ringing, reflections, pulse droop, etc. The type of distortion is quite different at various cable lengths and thus must be tested at three distances.



* Components added to normal circuit to create test circuit

Notes:

1. VCC should be adjusted to set pulse amplitude (VH,VL).
2. CLK should be adjusted for pulse width or idle time (TW, TI).
3. C1 should be adjusted to set rise time (TR).
4. Switching diodes can create a non-linear source impedance which can force idle amplitude to a maximum level because of transmission line reflections. Diode capacitance must be low to allow independent control of rise time.
5. R1 should be adjusted to set output impedance at 100KHz.
6. C2 should be adjusted to set frequency at which output impedance is down 3dB.
7. L1 may be added to adjust pulse droop if desired.
8. L2 may be added to increase pulse overshoot if desired.

Figure 3-9: Input Test Circuit Example

3.4.4 Input Filtering

There are several ways in which a waveform at a device input may be distorted. Externally generated electromagnetic interference (EMI) may affect the signal received by the device. The reactive portion of a device's output impedance will cause some waveform distortion (ringing, reflections, etc) generally in the 2 to 10 MHz range. Also, an input circuit using transformer isolation is likely to generate some distortion in this frequency range due to parasitic reactances.

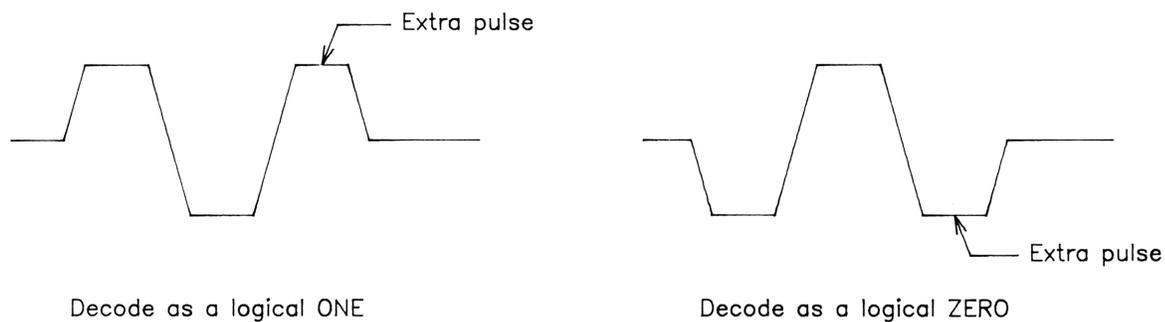


Figure 3-10: Extra Pulse Filtering

To enhance reliable signal detection, it is strongly recommended that some form of filtering be used to reject this distortion. In addition, the input shall ignore extra pulses as shown in figure 3-10.

3.5 Interface Cable Specifications

The interface cable between any two devices is a two wire balanced type cable of no more than 100 meters. A cable which is permanently or uniquely connected to a device output shall not be considered part of this 100 meter constraint. Thus, the distance between two devices may be lengthened arbitrarily if the device has a cable permanently or uniquely attached (i.e. dedicated cable) and meets the input/output specifications as measured at the end of the unique cable.

Throughout this section, references to cable specifications are to be interpreted as the entire link between one device's output terminals and another device's input terminals. If 2 or more cables are connected together between devices, the total length of all cables connected must meet the specifications listed in this section.

3.5.1 Cable Type

For distances less than 10 meters, there is no restriction on the type of two wire cable used. Therefore, a very low cost cable may be used as long it meets the specifications. Longer distances, however, require a shielded twisted pair. The shield prevents the characteristic impedance from being affected by environment or application (coiling, conduit installation, etc), and also helps balance EMI noise pickup onto each conductor to limit differential mode noise. The conductor twisting also helps balance noise pickup. The shield may be left unconnected or may be connected to earth ground provided the connection is made only at the input terminals of a device. If the shields of more than one interface cable are connected together, then grounding may only occur at one input terminal. The shield shall not be connected to device common if it is not also connected to earth ground.

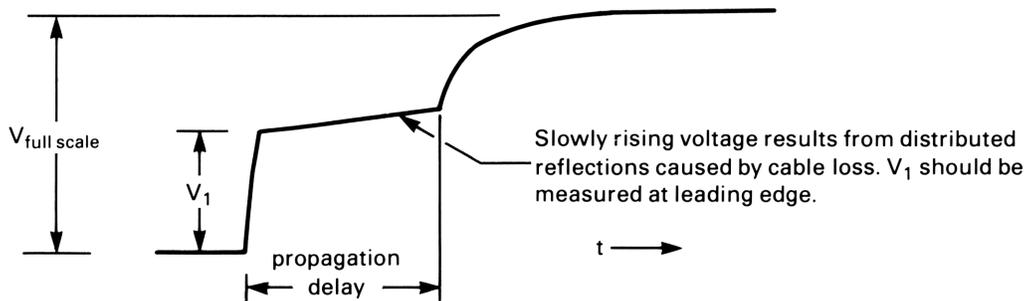
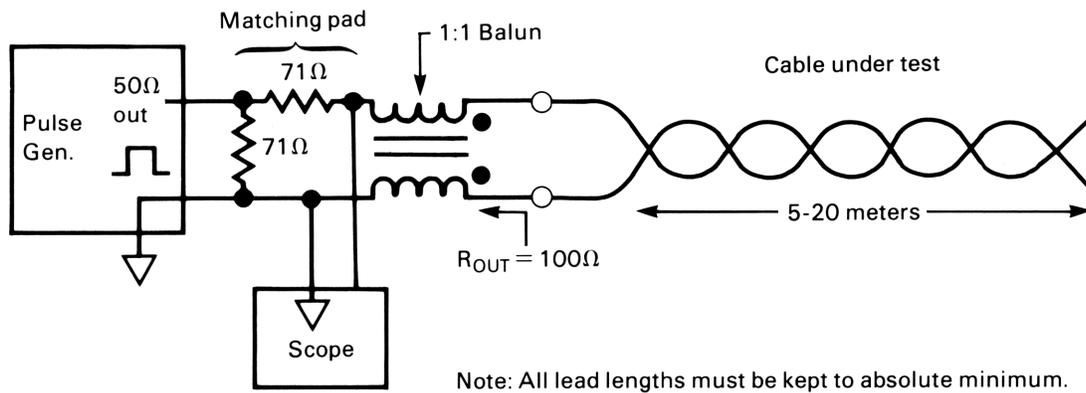
3.5.2 Characteristic Impedance (Z_0)

Characteristic impedance control is necessary to limit transmission line reflections. Reflections are less significant at short distances than they are at long distances, but they cannot be ignored. Z_0 requirements:

$$Z_0 = 100 \text{ ohms} \pm 20\% \quad (\text{cable length} < 10\text{m})$$

$$Z_0 = 100 \text{ ohms} \pm 10\% \quad (\text{cable length} > 10\text{m})$$

The characteristic impedance is measured using Time Domain Reflectometry techniques or equivalent. A suggested test procedure is given in figure 3-11.



Procedure:

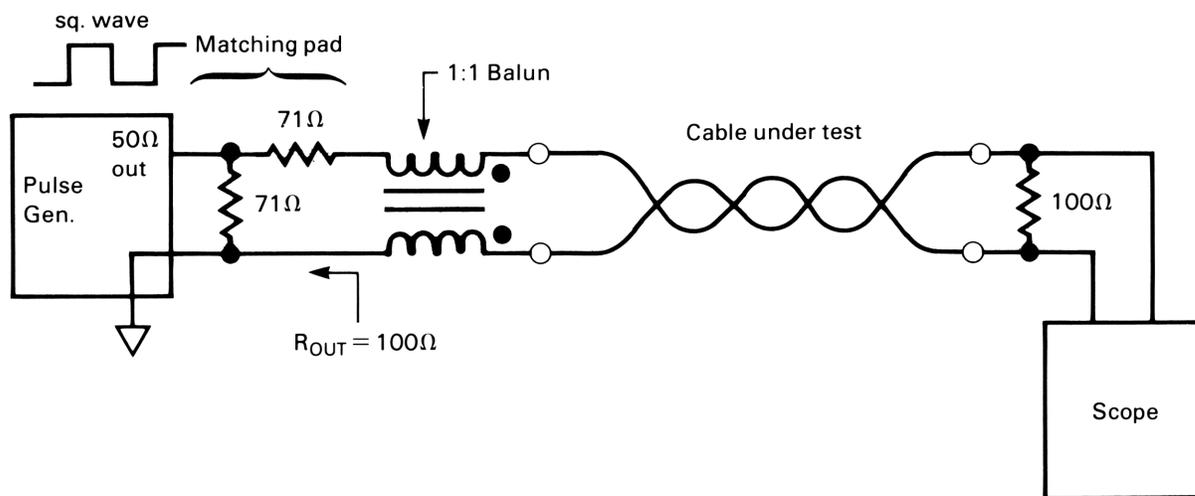
1. Adjust pulse width to be much greater than propagation delay.
2. With cable removed, adjust pulse amplitude for full scale deflection on scope.
3. Check test circuit output impedance by replacing cable with a 100 ohm resistor and verifying 1/2 full scale deflection.
4. Connect cable under test and measure V_1 .

$$Z_0 = \frac{(V_1 * R_{out})}{(V_{full\ scale} - V_1)}$$

Figure 3-11. Characteristic Impedance Test Procedure

3.5.3 Cable Rise Time

The 10% to 90% cable rise time must not be more than 200 nanoseconds when measured per figure 3-12 or equivalent. The cable(s) under test should be configured for the total length to be tested because rise times increase exponentially with distance (not linearly).



Procedure:

1. Measurement shall be made at the 100 ohm termination with a a balanced differential oscilloscope.
2. Make pulse width long enough so that all reflections and transients have time to settle to a steady state value.
3. Adjust the steady state voltage level for full scale deflection on scope.
4. Measure 10-90% rise time relative to the steady state values.

Figure 3-12. Cable Rise Time Test

3.5.4 Cable Loss

Cable losses are caused by series and shunt losses and are frequency dependent, being worst at high frequencies. The main result is to slow cable rise time (along with dispersion effects). Therefore, the rise time specification is sufficient to adequately specify high frequency losses. However, the rise time is specified relative to a steady state or DC level. For an open circuit termination, the steady state cable output will equal the source voltage. However, for a finite termination resistance, the cable DC series resistance will cause a loss for which the designer of an HP-IL input circuit must compensate (refer to figure 3.13).

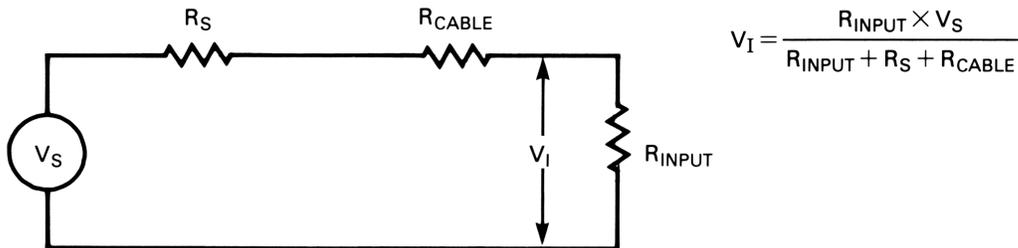


Figure 3-13. DC Cable Losses

The total DC cable resistance between two devices must not be more than 7 ohms on each conductor. For a 100 ohm output and input impedance, this would result in a 46.7% steady state attenuation, compared to 50% attenuation for a lossless line.

3.6 Isolation Requirements

Both signal lines of the input and output terminals must be isolated from earth (chassis or safety) ground and device common (for devices without earth ground) as follows:

Capacitance < 100 pf

Resistance > 10 MegOhms (40 deg C, 80% rel. humidity)

Voltage Breakdown > 500 V dc

3.7 Electromagnetic Compatibility (EMC)

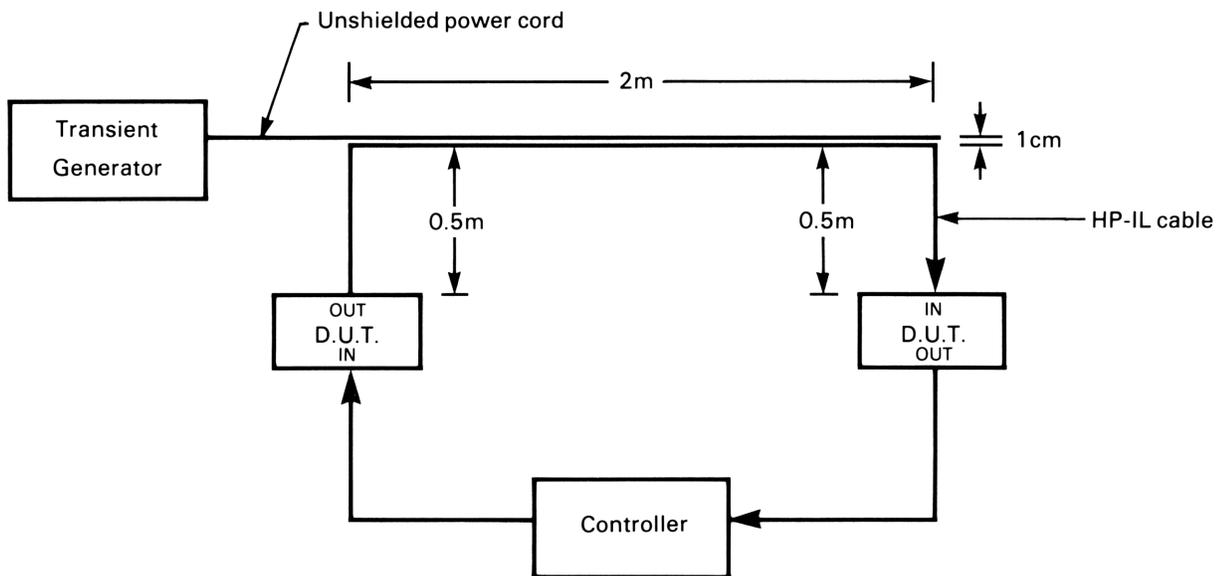
The EMI emissions of the interface shall be in compliance with VDE and FCC class B radiated and conducted interference requirements for cable lengths under two meters. The interface shall comply with class A requirements for cable lengths greater than two meters.

Susceptibility to EMI in HP-IL devices must be limited as much as possible because the HP-IL system may be used in environments which are subjected to many sources of EMI. One of the most severe sources is from transients on power lines in close proximity to the HP-IL cable. For more information on this subject, refer to General Electric's Transient Voltage Suppression Manual*.

It is assumed that HP-IL devices will perform error checking and that many devices will perform error handling and recovery such that only a small number of errors would require additional action from a user. Therefore, the specification allows for a non-zero error rate that insures reliable operation in most normal environments.

Because of the balanced nature of the interface cable, most of the EMI picked up by the cable will be in the form of a common mode signal. Therefore, this section will primarily test for the common mode rejection ability of an HP-IL device. While not explicitly specified, this will require the devices to maintain a reasonable amount of balancing, isolating, and/or shielding within the input and output circuits. Refer to figure 3.14 for test procedure.

* Transient Voltage Suppression, 3rd ed. (New York: General Electric Company, 1982)



Procedure:

1. Transient Generators:
 - Schaffner 222 or equivalent (5ns rise time)
 - Velonex 360 or equivalent (1.25MHz damped sine wave)
2. Identical units of the Device Under Test (D.U.T.) should be located as shown. If only one unit can be tested, it must be tested in both positions shown to test both the input and output circuits, and the controller's input and output circuits must comply with this section.
3. The D.U.T.s shall be tested on and off ground planes, and with AC power source connected if applicable.
4. The controller shall generate an average of more than 1000 frames per second. Error checking must be performed and the number of errors must be counted.
5. An oscilloscope should be used to verify the actual frame rate prior to the test. The oscilloscope should not be connected to the HP-IL interface during testing.
6. Both common and differential mode line transients should be applied to the power cord.
7. Because the line transients and the HP-IL frames are not synchronized, the test shall be run for more than five minutes and the total number of errors averaged for that interval.
8. Line transient repetition rate shall be set to maximum: 30Hz on Shaffner 222, 100Hz on Velonex 360.

Figure 3-14. Line Transient Test

Table 3-2: EMI Test Limits

Transient Generator	Amplitude	Normalized Error Rate* (max)
Schaffner 222 (rise time set to 5ns)	500V 1000V	2.0 E -7 2.0 E -6
Velonex 360 (IEEE 472)	500V	5.0 E -7

* Normalized error rate:

$$\frac{\frac{\text{Total number of errors}}{\text{test time (seconds)}}}{\text{Frame rate (Hz)} * \text{transient rate (Hz)}}$$

For example, a 1KHz frame rate, a 30Hz transient rate on the Schaffner 222, and a 1000V test amplitude, the maximum allowable number of errors in a 5 minute test would be 18.

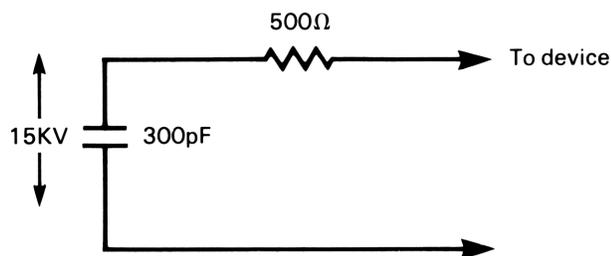


Figure 3-15. ESD Equivalent Circuit

3.8 Electrostatic Discharge (ESD)

HP-IL devices shall not experience any permanent failures as the result of a 15KV potential applied to any point on the interface, including the cable conductors. The equivalent circuit is shown in figure 3-15. This test represents a situation where a statically charged person may touch the interface, or may touch a device on the interface which couples the resulting discharge onto the interface cable.

It is strongly recommended that no temporary errors (data loss, resets, etc.) occur when this test is performed.

4. MECHANICAL SPECIFICATIONS

As mentioned previously, due to the special requirements of HP-IL a special connector set is used. Devices may either use a panel connector or cables affixed directly to the device through strain reliefs (figure 4-1). It is strongly recommended that devices use panel connectors in cases where a choice is available. Cable connectors (figure 4-2) are designed to be non-invertible and non-reversible and have a positive detent both in panel connectors and in each other for running cable "splices". In short, they are intended to be as foolproof as possible.

The connector contacts were chosen for high reliability and long life. The connector body is of molded polycarbonate while the integral strain relief is of PVC.

For applications where the distance from one device to the next is 10 meters or less, relatively inexpensive "zip" cord cable may be used. The wires are 24 AWG stranded (26 X 38 AWG) individually tinned copper. The wires are spaced 0.060 inches center-to-center with a PVC jacket, 0.065 X 0.130 inches. A polarity rib must be visible on the jacket.

For applications requiring longer distances up to 100 meters, cable which satisfies the more stringent electrical specifications given in the previous chapter must be used, that is, shielded twisted-pair cable whose characteristic impedance is 100 Ohms plus or minus 10%. Refer to chapter 3 for electrical characteristics of the cable.

Mechanical Specifications

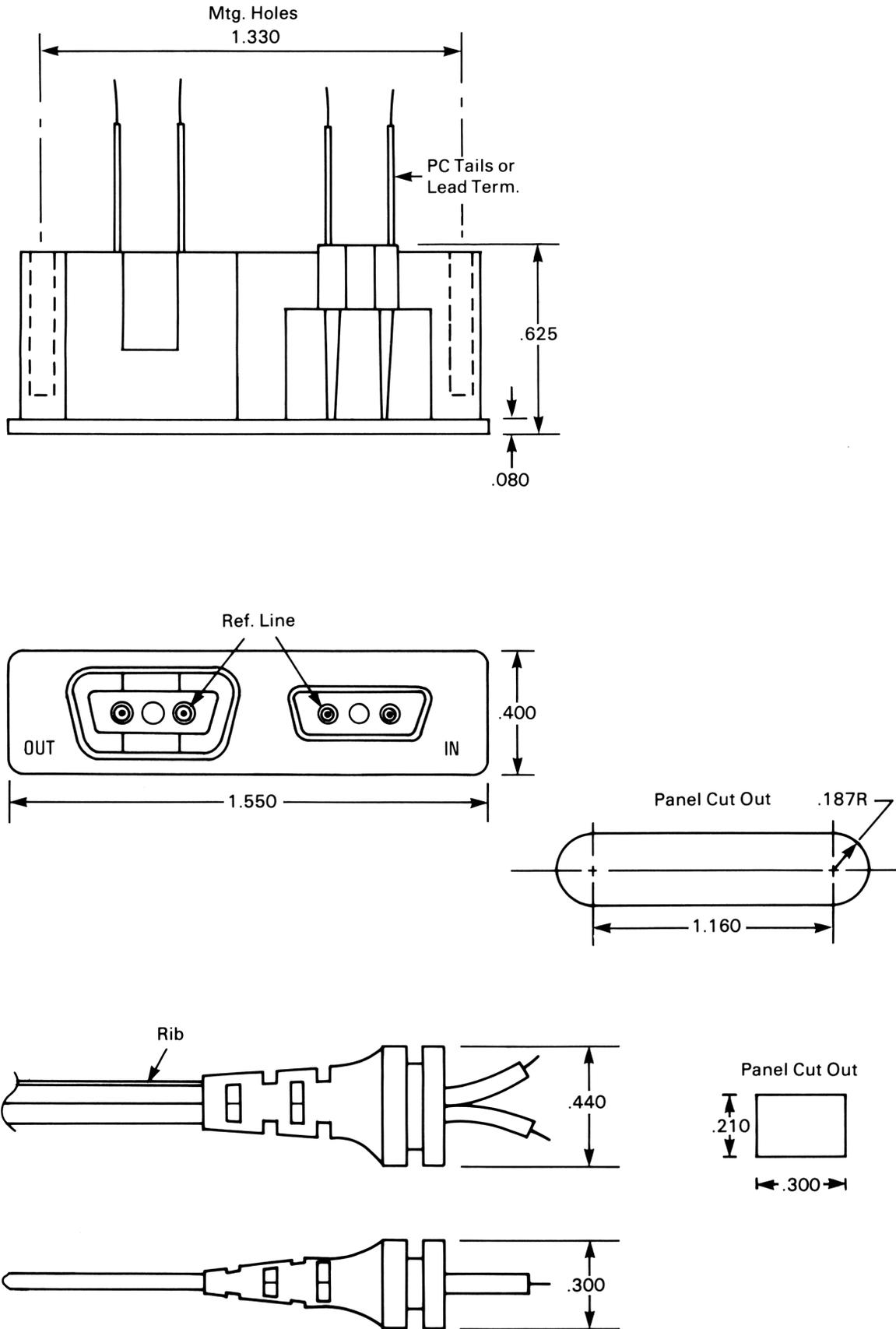


Figure 4-1. Example Device Connectors

Mechanical Specifications

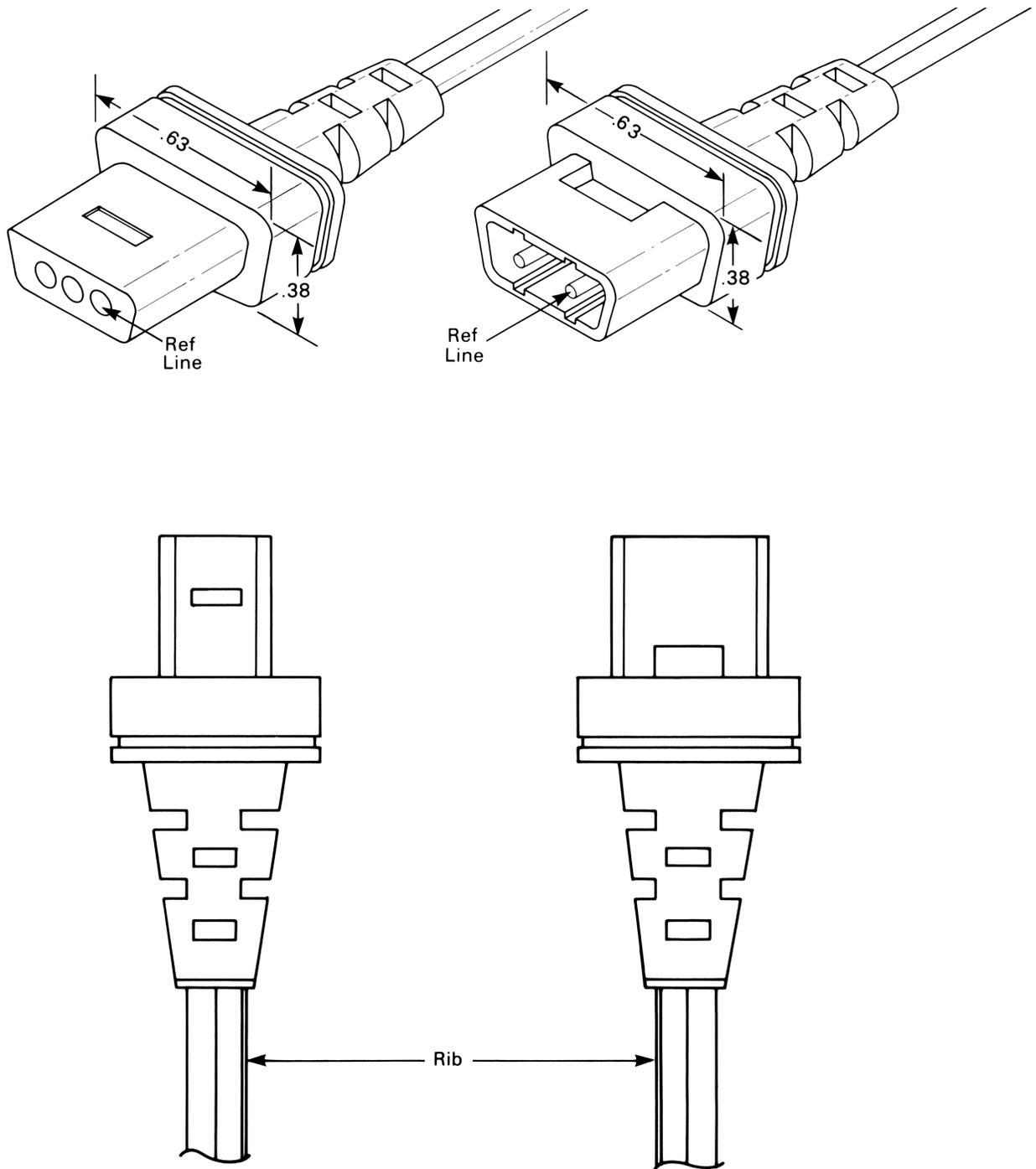


Figure 4-2. Cable Connectors

5. SYSTEM GUIDELINES

5.1 System Compatibility

This interface system offers a wide range of capability from which to choose the appropriate interface functions to fit different applications. Within most interface functions a number of options are available. In addition, the designer has freedom to select all the device dependent capabilities contained within the device functions.

It is the responsibility of the designer to define the complete capability of a device (interface function choices and related device dependent interactions) so that the end user of the device can efficiently interface and program the device for appropriate system applications.

Also, devices designed to this interface system may have a wide range of capability relative to their ability to communicate over the interface. This document does not cover the operational characteristics of devices, only the functional, electrical, and mechanical capabilities of the interface system.

The burden of responsibility for system compatibility at the operational level is on the user. The user must be familiar with all device characteristics interacting with the interface system (for example, device dependent program codes, output data format and codes, etc.).

In particular, compatibility will be greatly enhanced if the ASCII code is used by all devices for device dependent messages whenever possible. In addition, it is recommended that device dependent commands be avoided unless absolutely necessary. Simple ASCII programming codes or ASCII escape sequences are preferable for the implementation of simple "friendly" controllers.

5.2 System Configuration

If all devices on the loop use only one byte addresses, the maximum number of devices in an HP-IL system is 31. If all devices use extended addressing, the maximum number is 961.

In general, an interface system shall contain one or more devices containing at least one T function, one L function, and one C function.

If a T function includes the use of the local message ton and at least one L function includes the use of the local message lon, a system may be operated without a C function while the ton message is true in one and only one T function and the lon message is true in one or more of the L functions. The ton and lon messages are normally provided by local switches. This permits rudimentary, manual system operation in which one talker sends data continuously to one or more listeners.

All system configurations containing more than one controller must satisfy the following conditions: (1) There shall not be more than one C function in a system which is sending the scl local message true at any time. (2) Every controller in the system shall be able to pass and receive control of the interface.

Because of the serial nature of HP-IL, all devices must be powered on in order for the system to function. If desired, the designer may implement the interface circuitry such that when the device is powered off, power remains on for the interface circuitry to act as a simple repeater.

5.3 Address Assignment

Normally, a device will be assigned a single talk and a single listen address to perform the essential tasks. It may be useful to design a device with multiple talk (or listen) addresses to facilitate system requirements. A device could be assigned two talk addresses, for example, one to output raw data, the other to output processed data. If this is necessary, it is recommended that two-byte addressing be used so that limited one byte addresses will still be available.

When devices are powered on they may either respond to no address at all or they may respond to a default or switch selectable address. Responding to no address reduces the chances of two devices having the same address (which could happen if a device is reset during normal operations). When the AAU (auto address unconfigure) command is received, devices must respond to a preset or switch selectable address. Note that some devices require the AAU message to be sent before default addressing may be used.

It is important to note that the auto addressing messages have no effect on the T or L interface functions. All messages that affect the T and L functions are defined in chapter 2 and do not include any auto address messages. Therefore, if a device is in an addressed state when the AAU message arrives, it will remain in that state and respond accordingly to messages which follow. Normally, the messages which follow will assign new addresses. The act of assigning addresses to particular devices is distinct and separate from the act of addressing a particular device to talk or to listen. Refer to the appropriate sections of chapter 2 for more information.

A device that contains the T function may be assigned any value for the least significant five bits of its MTA (my talk address) message code other than 11111. This code, defined as UNT, is provided for the controller to return all devices to the talker idle state. Two or more T functions shall not be assigned the same value for these bits unless different secondary addresses are assigned. A device that contains both a T and an L function should have the same value for the least significant five bits of both the MTA and MLA (my listen address) messages.

A device that contains an L function may be assigned any value for the least significant five bits of its MLA message code other than 11111. This code, defined as UNL, is provided for the controller to return all devices to the listener idle state.

A device that contains extended talker or listener functions may be assigned any value for the least significant five bits of its MSA (my secondary address) message code other than 11111. Two or more devices with the same value of their MTA codes may not have the same value for MSA codes. If a device has both extended talker and listener functions, the lower five bits of the MTA and MLA codes should be equal and both functions should use the same MSA code as well.

In general, and particularly when auto addressing or auto extended addressing is used, the T and L function within a specific device will be assigned and will use the same address. Auto multiple addressing is intended specifically for those cases where there are multiple devices or addressable device functions within one mainframe. A block of secondary addresses is reserved by the device and these are assigned to the various T and L functions as the designer may choose.

While it is possible to mix devices which use simple addressing, extended addressing, and multiple addressing on the same loop, this should only be done with a good deal of caution and thought for the interactions of the various addresses. For example, the controller will need to assign addresses to the devices with extended addressing before assigning addresses to multiple addressing devices. Also, devices may be built that can perform either two byte addressing or one byte addressing (depending on the order that the controller assigns them). The controller must also exercise care such that no two talker functions are assigned the same primary and secondary address combinations (no secondary address is equivalent to having all secondary address codes assigned).

5.4 Asynchronous Operations

The normal loop handshake allows for only one frame to be in transit around the loop at any given time. For three special tasks, however, the controller may choose not to use this standard message handshake. These cases are: loop power-up or reset; integrity check or SRQ detect during slow handshake cycles; SRQ detect during quiescent periods.

Due to the serial nature of the loop and the presence of devices of various speeds, multiple frames may "stack up" at one device and frames may be lost if the normal loop handshake is not performed. For this reason, it is important to view asynchronous loop operation as a separate, non-standard mode of operation in which only certain, carefully defined tasks are accomplished. The burden of handling the possibilities mentioned here rests with the active controller and only the active or system controller may enable or disable the asynchronous modes of operation. This may affect the controller designer, the system programmer, or both.

5.4.1 Power up and Error Recovery

If the system controller chooses to power up the interface or reset it from an error condition, it may do so by sourcing the IFC (interface clear) message. Only the system controller may send the IFC message. It may do so at any time it chooses with the understanding that it may cause data loss if performed during data transfers.

Generally, sourcing the IFC is an asynchronous operation and should always be performed in the following manner. First, the system controller should send the IFC message and then wait for it to return. If it does not return within some relatively long time period (100 milliseconds, for example), the system controller should send the IFC again and wait, repeating this sequence until the IFC returns. It is possible that other frames will be received by the system controller before the IFC returns. Because these frames may have been sourced by a device that has become idle (due to the IFC message), they must be destroyed by the system controller to prevent their endless circulation around the loop. When an IFC returns, the system controller should then source one and only one RFC message to complete the handshake. All frames received by the system controller before the RFC returns must also be destroyed. When the RFC returns the system controller knows that all interface functions are initialized and that the loop is free of extraneous frames. If the RFC message does not return after a long time (say, 10 seconds), an error has occurred and the entire sequence should be repeated.

The IFC sequence is not intended to preserve message frames on the loop so the system controller need not concern itself with the destroyed frames. All system controllers must be able to execute this sequence in order to properly power up the interface.

5.4.2 Loop Integrity Check

During very slow loop operations, the controller may source an IDY (identify) message asynchronously to verify loop integrity or to check for service requests (for improved response time). The IDY is immediately retransmitted by all devices and will return to the controller relatively rapidly with service request and, optionally, parallel poll information. If the IDY message does not return within a reasonable amount of time, then the loop may be broken and an error condition exists.

The active controller may send the asynchronous IDY during any loop operation, but great care must be exercised during the RFC handshake of the CMD-RFC sequence. Since the CMD-RFC handshake cycle is the only case in which loop devices will normally have two frames waiting for interpretation (the CMD and the RFC), asynchronous IDY frames sent before the RFC frame returns risk destroying the RFC or the IDY frame (or both) because of frames stacking up. Therefore, IDY frames should not be sent by the active controller after RFC frames unless error recovery techniques (such as sending the RFC again) are used.

In cases other than the CMD-RFC sequence discussed above, the asynchronous IDY frame will not cause frames to be lost and the IDY will very likely overtake and pass whatever other message is presently on the loop. The functional specifications require that this other message not be affected in any way by the passage of the IDY. Loop operation should continue without any loss of data. Provided that the controller only sources one asynchronous IDY at a time, there will be no problems with frames being lost by the loop devices. When the IDY returns, the controller is then free to source another IDY if desired.

5.4.3 Asynchronous Service Requests

When the loop is quiescent, devices do not have a way to notify the controller of requests for service. Therefore, for the controller to know when service has been requested, it must either periodically send IDY frames, or it may enable devices to source their own asynchronous IDY frames (with the SRQ bit set).

To enable devices to source their own IDY frames, the controller sends the EAR command (enable asynchronous requests), the RFC, and then simply stops sending messages. When devices receive this command, they enter a mode in which, if they have the capability and require service, they may source their own asynchronous IDY. This message will not be retransmitted by the active controller, but may cause it to resume normal operation which includes handling the service request.

Because of the possibility of destroyed frames, the controller may only source UCG (universal command group) messages while the asynchronous request mode is in effect, and all UCG messages (except EAR and LPD) disable this mode and return the loop to normal operation. Note that parallel poll information contained in IDY messages from other devices is not accurate since they may not have travelled completely around the loop.

Devices are enabled to source their own IDY frames immediately after receiving the EAR command, and therefore, the controller must not require the RFC following the EAR command to return (it could be destroyed by the IDY's).

The controller should return the loop to normal operating mode in the same manner as the system controller handles the IFC sequence due to the asynchronous nature of this mode. The UCG command is sent first which disables asynchronous request mode. NOP is recommended as it does not otherwise modify the state of the interface (other UCG commands may be used if their effects are desirable). Other IDY messages may return before the CMD is received. They should be ignored (not retransmitted). If the UCG command does not return after a period of time, it should be sent again, and so on, until it does return (there is a possibility that the UCG command could be destroyed by frame "stack up"). The RFC can then be sent. One or more additional IDY messages may possibly arrive before the RFC returns. They should be ignored as were the earlier ones. When the RFC returns, the loop is cleared and has been returned to its normal, synchronous operating state. The controller can now perform normal operations such as servicing the request. The same sequence should be used to bring up a loop which has been powered down with the LPD command.

It is easily seen that the implementation and programming of controllers with asynchronous capabilities is not a simple task. It should only be attempted by designers and programmers who have a complete and total understanding of loop protocol.

5.5 Operational Sequences

To understand more completely the operation of an HP-IL system, frame sequences of several typical operations are shown. The sequences are not intended to be exhaustive, nor do they necessarily represent the only way to accomplish the given task. They are included as recommended sequences for performing the given operation. The messages listed in the column on the left represent those sent by the controller. The column on the right shows the messages sourced by the talker.

Initial System Power On

IFC The system controller initially sends the IFC
 IFC (Interface Clear) command at regular, slow intervals
 IFC stopping any and all frames received until an IFC
 . returns. When an IFC frame returns, the controller
 . knows that all devices are powered up and are properly
 IFC sending and receiving frames. The system controller
 then sends one and only one ready for command message
 RFC to complete the handshake and waits for it to return
 also stopping any other frames received. When the RFC
 returns, the controller knows that the loop devices
 have properly completed interface initialization and
 that no extra frames are on the loop. If the RFC does
 not return, the sequence should be repeated.

Talk-only, Listen-only System Power On

DAB1 The talker immediately sends its first data byte and
 . continues to send it at slow, regular intervals until
 . it returns, indicating that the other devices on the
 DAB1 loop are powered up and are properly retransmitting or
 receiving frames. The talker then sends the rest of
 DAB2 its data at normal speed. No EOT message is sent in
 DAB3 a talk-only system since it would circulate endlessly.
 . Transmission continues in this manner until the local
 DABn message ton is turned off.

Data Transfer

UNL The controller first inhibits any previous listeners
 RFC with the UNL (Unlisten) command and enables a listener
 LADn with the appropriate listen address command. More
 RFC than one listener can be enabled, if desired. The
 TADn talk address command disables any previous talkers and
 RFC enables one device to send data when it receives the
 proper ready message. The controller now sends the
 SDA Send Data ready message which the talker replaces on
 DAB1 the loop with its first data byte. As each byte
 DAB2 returns, the talker sends the next data byte of its
 DAB2 message. When the talker has no more data ready to
 . send, it sources the ETO (End of Transmission, OK)
 . message. ETO indicates to the controller that all
 . data frames sourced by the talker returned without
 . error. The controller replaces the ETO message with
 . its next interface message.

Interruption of Data Transmission

Assume the talker is sending its seventh data byte. The device halting the data transfer holds the talker's byte, and replaces it with the NRD (Not Ready for Data) message. The NRD signals the talker to stop its data transmission immediately. The talker returns the NRD frame to the halting device which retransmits the held data byte. When the talker receives its data byte, it sends an end of transmission message to allow the controller to perform other operations (such as handling a service request). If the talker is again directed to send data, the transmission will continue at the point of interruption (unless directed otherwise in a device dependent manner).

DAB7
 NRD
 NRD
 DAB7
 ETO
 .
 .
 .

Serial Poll

UNL
 RFC
 TAD1
 RFC
 SST
 DAB
 .
 ETO
 TAD2
 RFC
 SST
 DAB
 .
 ETO
 TAD3
 .

The controller first sends the unlisten command to disable all previous listeners.

The first device is addressed to talk and asked to send its status with the Send Status ready message. The talker replaces the SST frame with its status message (one or more bytes of data) followed by ETO.

The controller continues by addressing the second device to talk. This same sequence (TAD,RFC,SST) is repeated for each device on the loop. Devices that currently have a need for service will respond to the serial poll with bit 6 of the first status byte set. If the SST frame returns to the controller, it knows that the addressed device does not have the capability to request service or to respond with its status.

Control Passing

TAD
 RFC
 TCT
 .
 .
 .

The currently active controller sends the talk address of the device to which it will pass loop control followed by the TCT (Take Control) ready message. When the TCT message is received by the talker, itive takes over active control of the loop by replacing the TCT with its first interface message. If, instead, the TCT returns to the previous controller, the talker has not assumed control of the loop and the previous controller must resume control.

Parallel Poll Configuration and Operation

- PPU The controller first may send the PPU (Parallel Poll
RFC Unconfigure) command to disable all previous parallel
 poll response assignments.
- UNL The controller sends the Unlisten command to prevent
RFC unwanted devices from reacting to parallel poll enable
LADn commands. The listen address of the device to be
RFC configured is sent next followed by the specific
PPE13 parallel poll enable command. In this case, the device
RFC is assigned to respond with a positive response by
 ORing a 1 into bit 3 of an IDY frame.
- UNL This same sequence (UNL,RFC,LAD,RFC,PPE,RFC) is
RFC repeated for each device to be configured for parallel
 . poll. Individual devices can be disabled by
 . substituting the PPD (parallel poll disable) command
 . for the PPE command in this same sequence.
- IDYØ After the devices have been configured, the controller
 may execute the parallel poll at any time by sending
 the IDY message. The frame will return with its data
 bits set according to the configuration commands and
 the devices' individual status bits.

Assign One Byte Addresses

- AAU The controller first sends the AAU (Auto Address
RFC Unconfigure) command to reset any previous address
 assignments and make all devices ready to receive new
 addresses. Then the controller sends the AAD (Auto
AAD1 Address) ready message. The first loop device accepts
 the lower five bits of the AAD message as its new
 address assignment and increments the AAD message
 before sending it on to the next device. This process
 (accept address, increment, send to next device)
 occurs at each device on the loop until the modified
AADn frame returns to the controller, indicating the number
 of devices on the loop. If the frame returned with
 address 31, there may be too many devices on the loop.
 To determine this, the controller should now send
 AAD3Ø. If it returns unchanged, there are exactly the
 maximum number of devices. If IAA returns, there are
 too many devices and improper operation may result.

Appendix A. Capability Subsets

R (Receiver) Interface Function

All devices must have the complete R function implemented.

D (Driver) Interface Function

All devices must have the complete D function implemented.

AH (Acceptor Handshake) Interface Function

All devices must have the complete AH function implemented.

SH (Source Handshake) Interface Function

All devices must have the complete SH function implemented.

C (Controller) Interface Function

Identification	Requirements
C0 No capability	Omit all states
C1 Basic capability	Implement CIDS, CACS, CSBS, CEIS, CEMS
2 System controller	Additional to C1; local messages sic, scl not always false
3 Respond to service requests	Additional to C1; add CSNS, CSRS (needs listener capability)
4 Pass, receive control	Additional to C1; add CTRS and optional term; (requires talker capability)
5 Parallel poll	Additional to C1
6 Enable and interpret asynchronous IDY frames	Additional to C1

T (Talker) Interface Function

Identification	Requirements
T0 No capability	Omit all states
T1 Send data	Implement TIDS, TADS, TACS, TAHS, TERS
T2 Send status	Implement TIDS, TADS, SPAS, TAHS, TERS
T3 Send device ID	Implement TIDS, TADS, DIAS, TAHS, TERS
T4 Send accessory ID	Implement TIDS, TADS, AIAS, TAHS, TERS
5 Talk-only mode	Additional to T1; local message ton not always false
6 Extended talker	Additional to T1-4; implement TPIS and TPAS (requires AA2 or AA3)

Talker capability consists of one or more of T1-4.

L (Listener) Interface Function

Identification	Requirements
L0 No capability	Omit all states
1 Basic capability	Implement LIDS, LACS
2 Listen-only mode	Additional to L1; local message lon not always false
3 Extended listener	Additional to L1; implement LPIS and LPAS (requires AA2 or AA3)
4 Not ready	Additional to L1; implement NIDS, NENS, NRWS, NACS

SR (Service Request) Interface Function

Identification	Requirements
SR0 No capability	Omit all states
SR1 Basic capability	Implement SRIS, SRSS, SRHS; needs T2 capability
SR2 Basic and asynchronous request capability	Implement all states; requires T2 capability

Appendix A Capability Subsets

RL (Remote Local) Interface Function

Identification	Requirements
RL0 No capability	Omit all states
RL1 Basic capability	Implement RIDS, RACS, LOCS, and REMS; local message rtl always false; needs listener capability
RL2 Basic capability with local lockout	Implement all states; local message rtl not always false; needs listener capability

AA (Auto Address) Interface Function

Identification	Requirements
AA1 Basic capability	Implement AAUS, AAIS, AACS
AA2 Extended addressing	Implement AAUS, ASIS, AWPS, AECS
AA3 Multiple addressing	Implement AAUS, APIS, AWSS, AMIS, AECS

All devices must implement one or more of AA1-3.

PD (Power Down) Interface Function

Identification	Requirements
PD0 Basic capability	Implement POFs, PONS, PUPS; pseudomessage edge always false
PD1 Complete capability; responds to power down command	Implement all states

All devices must implement one of PD0, PD1.

PP (Parallel Poll) Interface Function

Identification	Requirements
PP0 No capability	Omit all states
PP1 Complete capability	Implement all states; needs listener capability

DC (Device Clear) Interface Function

Identification	Requirements
DC0 No capability	Omit all states
DC1 Respond to universal device clear command	Implement all states; omit optional term
DC2 Respond to universal and addressed device clear commands	Implement all states; include optional term; needs listener capability

DT (Device Trigger) Interface Function

Identification	Requirements
DT0 No capability	Omit all states
DT1 Complete capability	Implement all states; needs listener capability

DD (Device Dependent Commands) Interface Functions

Identification	Requirements
DD0 No capability	Omit all states
DD1 Responds to one or more device dependent listener or talker commands	Implement all states; needs listener capability for DDL, talker capability for DDT commands

Appendix B. Message Glossary

B.1 Local Messages and Pseudomessages

- arq - Asynchronous request. To SR (service request) function. The device uses arq to signal the function to send the asynchronous IDY (identify) message to the controller to request service. This will only happen if the device has this capability and has been enabled to do this by the EAR (enable asynchronous requests) command. The arq message must go false as soon as the function enters the state which sources the IDY (ARSS). It may go true again upon return to the standby state (SRSS).
- edge - Wake-up signal. Pseudomessage to PD (power down) function. If the device has the full power down capability and has been powered down with the LPD (loop power down) command, the function will be in the power off state, even though the power switch will still be on. In this condition, any frame (actually, any pulse) on the loop will generate the edge message and cause the function to bring the device back to full power. The edge message is a pulse which must only last long enough for the device to completely power on to its initial state. When edge goes false, the device will then be ready for normal operation.
- fon - Force on. To PD (power down) function. Under certain conditions, devices which have the full power down capability may need to remain powered up even though they have received the LPD (loop power down) command. The fon local message causes them to go back to PUPS (power up state) rather than go to POFs (power off state). The active controller would use this feature to remain awake, while presently inactive controllers could allow themselves to power down normally. If a device was enabled to send asynchronous IDY messages to wake up the loop, it might also use this message to remain awake.

- fre - Frame error detected. To T (talker) function. All data frames sourced by the active talker should be error checked when they return. If an error is detected, then the fre message should be sent to the T function causing it to source the ETE (end of transmission, error) message.
- frtc - Frame transmission complete. Pseudomessage to D (driver) function. This message is generated by the encoder circuitry to signal the D function that it is done transmitting an entire frame and that the function can now return to the idle state to wait to begin transmitting another frame. This message should only be true until the function returns to the idle state.
- gta - Go to active. To C (controller) function. When the controller enables a device dependent transmission, it goes to the standby state to wait for completion of the transmission. While the controller is in its standby state, it may not source any frames. If it becomes necessary to source asynchronous frames such as IDY or IFC, the device may force the C function to become active (so that it is enabled to source frames). The gta message should only be true until the function returns to the active state.
- gts - Go to standby. To C (controller) function. After the controller has used the gta message to force the C function to become active during a device dependent transmission, it uses the gts message to return to the standby state to wait for the normal completion of the transmission. This message should only be true until the function returns to the standby state.
- hlt - Halt data transfer. To L (listener) function. This message is used by the device to cause the L function (if controller or enabled by the controller) to send an NRD (not ready for data) sequence to halt the data transfer. The hlt message is especially useful for listeners with limited input buffers.
- lab - Local abort. To SH (source handshake) function. This message is used for asynchronous operations to cause the SH function to abort the current handshake and become ready for another message. It would otherwise be "hung" waiting for the previous message to return before it could generate the new message. The lab message is only true until the function returns to the generate state.

- lfs - Last frame sent. To the T (talker) function. The device will send this message true when it detects that there is no more data ready to be sent. The T function will then source the ETO (end of transmission, OK) message unless the fre local message is also true.
- lon - Listen only. To L (listener) function. This message is usually controlled by a manual switch and serves to make the device an active listener even though it has not been addressed. This is only allowed in a special system configuration which has no controller called a talk-only, listen-only system.
- ltn - Local listen. To L (listener) function. This message is used by the active controller to make itself an active listener without sending out its own listen address, though it could do this if it wished. This message might be used to allow the controller to monitor a device dependent message transfer between other devices or it might also be used if the controller itself were the destination of the data. The ltn message will only be true until the function enters the active state.
- lun - Local unlisten. To L (listener) function. This message is used by the active controller to return the listener function to the idle state after it has been forced active by the ltn local message. This message will only be true until the function returns to the idle state.
- nfa - New frame available. To SH (source handshake) function. When nfa goes true, the device is telling the SH function that it has generated another frame for transmission and is ready to have it sent. The nfa message must go false before the function can become ready to source the next frame.
- pof - Power off. To PD (power down) function. This message is usually sent true by the "off" position of the device power switch. It causes an immediate transition from any state to POFS (power off state). This, in turn, causes all other functions to go to their power off states. This message remains true until the switch is moved to the "on" position.

- pon - Power on. To PD (power down) function. This message is a short pulse generated when the device power switch is moved to the "on" position. It causes the PD function to move from the power off state to PONS (power on state). The function remains in PONS until pon goes false. The pon message must last long enough to allow all device and interface functions to go from powered down to powered up and ready for normal operation. When pon goes false, the function enters PUPS (power up state).
- rdy - Ready. To AH (acceptor handshake) function. With this message the device indicates to the AH function that it is now ready to receive the next byte of the incoming message string. The rdy message allows transition to ACDS (acceptor data state) which transfers incoming frames to all the other interface functions and device functions. The device indicates acceptance of the message from the function by setting rdy false. As long as the device is busy (it is interpreting the current message or its buffer is full) the rdy message should remain false. This message, therefore, controls the handshake of messages into the device.
- rsv - Request service. To SR (service request) function. The device indicates to the SR function a need for service with this message. It causes the function to begin sending the SRQ (service request) to the controller when it has the opportunity. The rsv message is the same as bit D6 of the device's first status byte.
- rtl - Return to local. To RL (remote local) function. This message is usually generated by a button on the device which returns control of the device functions to manual controls on the instrument itself. The function has the optional capability to ignore this message with the LLO (local lockout) command. The message must not be always true; it will usually be a short pulse.
- scl - System controller. To C (controller) function. This message indicates to the function that it is the system controller, the only device permitted to send the IFC (interface clear) message to take control of the loop at any time. This message should remain true in one and only one device throughout the operation of the interface.

- sic** - Send interface clear. To C (controller) function. The system controller uses this message to cause the function to go from the idle state to the active state so that it can send the IFC (interface clear) message and take control of the loop. The sic message must be used at initial power on as well as any other time the system controller needs to asynchronously take control of the interface. The message should only be true until the active state is entered.
- sync** - Valid sync bit received. Pseudomessage to R (receiver) function. This message is generated by the decoder circuitry when it recognizes the beginning of a message frame from the loop. It causes the function to go from idle to RSYS (receiver sync state) where it will decide whether the message is for this device or not, and then take appropriate action. The sync message should only be true until the function enters RSYS.
- tlk** - Local talk. To T (talker) function. With this message, the active controller can make itself the active talker and source device dependent data without sending an SDA (send data) message although it could do this instead, if desired. The tlk message should only be true until TACS (talker active state) becomes true.
- ton** - Talk only. To T (talker) function. This message is normally controlled by a manual switch and causes the device to become the active talker even though it has not received its talk address. This is only permitted in a special system configuration which lacks a controller and is called a talk-only, listen-only system. The function will remain in TACS (talker active state) sending continuous data messages as long as the switch is in this position.

B.2 Remote Messages

- AADn - Auto address n. Auto address group, ready class, 101 100aaaaa. The controller sends AADn to assign simple (one byte) addresses to devices on the loop. The lower five bits represent a binary coded address number n which can range from 0 to 30 (31 is an illegal address, devices will not respond to this value). Each device accepts the incoming value n as its address, increments this value, and sends the modified AAD to the next device on the loop, which, in turn, does the same. Once a device has received its address in this way, it will no longer respond to any AAG (auto address group) message until after the AAU command is received or the device is powered off, then on again. The controller uses the address value which returns after going through each device around the loop to determine the number of devices, or to determine if there are too many devices.
- AAG - Auto address group. Ready class, 101 lxxxxxx. This mnemonic indicates the entire group of auto address ready frames, including simple address, extended and multiple address, and secondary address assignment frames. The controller uses these to assign addresses to devices on the loop in various ways.
- AAU - Auto address unconfigure. Universal command group, command class, 100 10011010. The controller uses this command to cause all devices to reset their address assignments. After an AAU, devices must respond to either address switches or a preset address. If loop devices already have addresses assigned, the controller must send the AAU message before assigning new addresses to those devices. Note that IFC (interface clear) does not affect address assignments in any way.
- ACG - Addressed command group. Command class, 100 x000xxxx or 100 101xxxx or 100 110xxxx. This mnemonic indicates that group of commands to which a device does not respond unless it is addressed as a talker or a listener, depending on the particular command. This group also includes the device dependent commands, DDLn and DDTn, as well as others.

- AEPn - Auto extended primary n. Auto address group, ready class, 101 101aaaaa. After the controller has assigned secondary addresses with the AESn message to a group of devices which can accept extended addresses, it uses the AEPn message to assign the same primary address to each device in the group. Devices do not modify this message, they merely accept the address assignment and send the message to the next device. Other devices do not respond to this message. After the AEP, the device is configured and can respond to its assigned secondary and primary addresses. AAU will reset the address assignment and ready the device to receive a new address. aaaaa represents the five bit binary address n, which can range from 0 to 30 (31 is an illegal address, devices will not respond to 31).
- AESn - Auto extended secondary n. Auto address group, ready class, 101 110aaaaa. AES is used by the controller to assign secondary addresses to extended addressable devices. The lower five bits contain the binary address n, which can range from 0 to 30 (31 is illegal). Each device accepts the value n as its secondary address, increments this value, and sends the modified message on to the next device. When the value reaches 31, no other devices respond and the message simply returns to the controller. The controller can then use AEPn to assign the primary address to this group of devices. Once configured, the devices can no longer respond to the AESn, so the controller can now send it out again to assign extended addresses to the next group of devices on the loop. The primary address for each group must, of course, be unique.
- AMPn - Auto multiple primary. Auto address group, ready class, 101 111aaaaa. AMP assigns primary addresses to all devices which use multiple addressing on the loop. The lower five bits represent a binary address n, which can range from 0 to 30 (31 is illegal). The controller sends the AMP message and each succeeding device accepts the incoming value as its new address, increments n, and sends the message to the next device, which, in turn, does the same. The value which returns to the controller indicates the number of multiple address devices on the loop. Following this, the controller sends the ZES command to each device so that it can reserve the proper sized block of secondary addresses. The device is then configured and can respond to its assigned addresses. AAU is necessary before devices will respond to new address assignments.

- ARG - Addressed ready group. Ready class, 101 01xxxxxx. Only talkers, listeners, and controllers may respond to this group of messages. Idle devices must ignore (retransmit) these messages. ARG messages include SOT (start of transmission), EOT (end of transmission), and NRD (not ready for data) subgroups and messages. With the exception of NRD, these messages do not normally travel all the way around the loop back to the sourcing device. In general, they serve a handshake function and the destination device replaces them with another message. At present, listeners do not respond to these messages, but may source the NRD message if enabled by the controller.
- CMD - Command. 100 xxxxxxxx. Commands are one of the major classes of loop messages. They control the operation of the interface functions of each device in a major way, and to a lesser extent, the device functions also. The active controller is the only device which may source command messages (except for asynchronous IFC by the system controller). Every command must be immediately followed by the RFC message to provide devices the opportunity to handshake, that is, to indicate they are ready to receive the next command. Commands are immediately retransmitted by all devices to minimize delay but a copy of the message is saved by each device to begin execution of the command.
- DAB - Data byte. Data or end class, 00x xxxxxxxx. Data bytes are the basic unit of the device dependent message transmission. This is the data which the interface system is designed to handle. The other messages are largely overhead for control purposes. These messages between devices may be coded in any way but it is strongly recommended that ASCII be used wherever possible for compatibility reasons. The data byte also contains the SRQ bit (C0) which devices may set to indicate to the controller a need for service.
- DCL - Device clear. Universal command group, command class, 100 00010100. DCL is sent by the controller to cause all devices which recognize this command to set their device functions to a preset state, whether they are addressed or not. This command does not affect the interface functions. The preset state is defined by the device designer and is normally the same as the power-on state.

- DDL_n - Device dependent listener command n. Addressed command group, command class, 100 101xxxxx. A device must be addressed as a listener in order to respond to any one of the 32 possible DDL commands. The particular effect of the specific command is designer determined but it must not directly affect any interface functions.
- DDT_n - Device dependent talker command n. Addressed command group, command class, 100 110xxxxx. A device must be addressed as a talker in order to respond to any one of the 32 possible DDT commands. The particular effect of the specific command is designer determined but it must not directly affect any interface functions.
- DOE - Data or end. 0xx xxxxxxxx. This major frame classification includes all the device dependent messages for the interface system. This is the data which is communicated from one device to another and for which the system was designed. DOE frames include the END bit (C1) to indicate an end-of-record condition without terminating the transmission, and the SRQ bit (C0) for devices to indicate a need for service to the active controller. These messages are sourced by the active talker and are received by the active listener(s) on the loop.
- EAR - Enable asynchronous requests. Universal command group, command class, 100 00011000. This command is used by the active controller to put all devices which have the capability in a mode where they can source their own IDY message (with service request bit set) to indicate a need for service to the controller. Normally, the controller is the only device to source IDY frames. After the controller sends the EAR - RFC sequence, it will allow the loop to go idle, until such time as an asynchronous IDY from one of the devices arrives or until the controller must perform some other operation. To disable the asynchronous request mode, the controller must send out a universal command which disables the asynchronous request mode and then resume normal operation. All universal commands except EAR and LPD disable the mode. However, it is recommended that controllers use the NOP command as it has no affect on the other interface functions. LPD does not disable the mode so that it is possible to have some device other than the active controller wake up a powered down loop.

- ELN - Enable listener not ready. Addressed command group, command class, 100 00001111. This command enables listeners to halt data transfers when necessary. Any device that is active to listen may respond to the EDN command.
- END - End data byte. Data or end class, 01x xxxxxxxx. The end byte is the same as a data byte except that bit C1 is set to indicate an end-of-record condition to the listener. This has no affect on the interface functions and the end byte is treated exactly the same as any other data byte. Most ASCII transmissions will indicate end-of-line with a CR, LF pair, for example, but binary data will probably need to use the END message for this function. The END byte does not terminate the transmission.
- EOT - End of transmmission. Addressed ready group, ready class, 101 0100000x. This is a subgroup including two messages, ETE (end of transmission with error) and ETO (end of transmission, OK). The active talker sources these messages to indicate the end of a data transfer to the controller. The controller replaces the EOT message with the next interface message. Listeners must ignore (retransmit) the EOT messages.
- ETE - End of transmission with error. Addressed ready group, ready class, 101 01000001. This message is sourced by the active talker to indicate to the controller that a data message sent by the talker has returned in error. This error checking capability is strongly recommended but not required. If the device does not perform error checking, it may not source this message. The controller replaces this message with its next operation on the loop, possibly an attempt to restart the transmission.
- ETO - End of transmission, OK. Addressed ready group, ready class, 101 01000000. This message is sourced by the active talker to indicate to the active controller that it is no longer actively sourcing data. Though not required, error checking is strongly recommended. If error checking is not performed, the talker must assume that no errors have occurred and end its data transfer with the ETO message. The ETO message does not return to the talker but instead the controller replaces it with the next interface message. Active listeners must ignore (retransmit) the ETO message.

- GET - Group execute trigger. Addressed command group, command class, 100 00001000. GET is a command used by the controller to cause all devices which are listener addressed to begin their particular device operation. This operation for each device is designer specified. The controller may use the GET command to start an operation in several devices as nearly at the same time as is possible given the loop architecture of the system.
- GTL - Go to local. Addressed command group, command class, 100 00000001. The controller uses this command to put all devices which are listener addressed under control of their local (front panel) controls. Programming data for the device will, in general, be ignored while the device is in this state, if it is received from the interface.
- IAA - Illegal auto address. Auto address group, ready class, 101 10011111. If the controller receives this message as a result of assigning addresses with the AAD message, there may be exactly the maximum or too many devices on the loop. Since devices do not respond to the IAA message, one or more devices may not have been assigned an address. To test whether all devices have been assigned an address, the controller should send the AAD30 message. If AAD30 is returned, then exactly the maximum number of devices are on the loop and the controller may begin normal operations. If IAA is returned again, then too many devices are present and more than one device may respond to a particular address. Before normal operations may begin, all extra devices must either be removed from the loop or assigned to addresses from an unused address range. For example, the extra devices can all be assigned address 30 by repeatedly sending AAD30 until returns unchanged.
- IDY - Identify. 1lx xxxxxxxx. This major classification of messages is used by the controller to perform parallel poll or to check for service request. Bit C0 is set by devices which need service. If the controller has configured devices to respond to parallel poll, these devices set designated data bits in the IDY message as it passes through the device. With this capability, the controller can rapidly identify which device needs service. During normal operations IDY messages may only be sourced by the active controller, but other devices can be enabled to send IDY frames (with the service request bit set) with the EAR command.

- IEP - Illegal extended primary. Auto address group, ready class, 101 10111111. This message is this same as the AEP message except that the address value is 31, an illegal value. Devices will not respond to this message. Furthermore, since AEP is not incremented by the loop devices, it will not be received by the controller and it is included here only for consistency.
- IES - Illegal extended secondary. Auto address group, ready class, 101 11011111. Extended address devices receive their secondary address assignments via the AES message, which they accept, increment, and send on to the next device. When the address value reaches 31, it is defined as the IES message and is no longer accepted by other devices, which pass it unchanged back to the controller. The controller then assigns primary addresses with the AEP message, which is not incremented. Only devices which have received the AES message (and have not yet received the AEP message) will respond. After receiving both secondary and primary addresses, devices will no longer respond to AAG messages and the controller may configure the next group of devices. If the controller generates the IEP message internally and the AES message returns incremented, there are too many devices on the loop.
- IFC - Interface clear. Universal command group, command class, 100 10010000. IFC may be sourced only by the system controller. It may be sent at any time to take control of the interface system. IFC resets all talker, listener, and controller functions on the loop to their idle state, but does not affect any other interface or device functions. IFC also must not affect the parallel poll or address assignment.
- IMP - Illegal multiple primary. Auto address group, ready class, 101 11111111. The controller uses the AMP message to assign primary addresses to those devices which have multiple address capability. If there are exactly the maximum number or too many devices of this type on the loop, the AMP message will be incremented to 31, which is defined as the IMP message. No further devices will respond and the message will return to the controller. The controller should send AMP30 at this point. If it returns unchanged, the loop has exactly the maximum number of devices. If it returns modified, there are too many. The controller should signal an error condition and not attempt normal operations.

- LADn - Listen address n command. Listen address group, command class, 100 001aaaaa. This is the command that the controller uses to cause a particular device to become the active listener, that is, able to receive and interpret data messages from the loop. The lower five bits represent a binary address n, which can range from 0 to 30. 31 is an illegal address which causes all listeners to go to the idle state. It is called the unlisten command. For devices which use a two byte address, LAD provides the primary address only. The MSA command must be received to make these devices active listeners. Multiple LAD messages will activate multiple listeners.
- LAG - Listen address group. Command class, 100 001xxxxx. This group of commands includes all the listen address commands and also the unlisten command (address 31). Secondary address commands are in a separate group; only primary listen addresses are included in LAG.
- LLO - Local lockout. Universal command group, command class, 100 00010001. With this command the controller can cause all devices which respond to this command to lock out, or not respond to, their return to local control buttons on the instruments. This will prevent an operator from changing a device's control settings inadvertently at a critical time.
- LPD - Loop power down. Universal command group, command class, 100 10011011. The controller uses this command to place the loop, or rather all devices which respond to this command, in a power-down state to conserve power. The controller remains powered up so that it can wake up the loop at a later time and continue normal operations. If the controller has enabled the asynchronous request mode prior to sending the LPD, other devices with the proper capability can also initiate the loop wake-up sequence.
- MLA - My listen address. Listen address group, command class, 100 001mmmmm. This is the particular LADn command which happens to match in the least significant five bits with the address (primary address in the case of devices that have a two byte address) which is assigned to this specific device. This command causes the device to become the active listener and able to receive device dependent messages. In the case of devices which require two byte addresses, the MSA code is also required before the device becomes active.

- MSA - My secondary address. Secondary address group, command class, 100 011mmmmm. This is the particular SADn command which matches the secondary address assigned to this specific device. This command causes the device to become the active talker or to become an active listener. This command must immediately follow the MTA or MLA command.
- MTA - My talk address. Talk address group, command class, 100 010mmmmm. This is the particular TADn command which matches the talk address (primary address in the case of devices that have a two byte address) which is assigned to this specific device. When the device receives MTA, it becomes the active talker on the loop, and, when enabled, will source device dependent data on the loop. For devices which have a two byte address, the MSA command is also required before the device becomes addressed to talk. This command causes the previous talker to become idle.
- NAA - Next auto address. Auto address group, ready class, 101 100nnnnn. This mnemonic represents the incremented auto address message that a device sends on to the next device on the loop. The value of the lower five bits might range from 1 to 31, depending on the value of the AAD message before incrementing. The value 31 is also called IAA, values less than 31 are also called AAD.
- NES - Next extended secondary. Auto address group, ready class, 101 110nnnnn. The NES frame is the incremented secondary address assignment which devices send to the next loop device. It is used by devices with extended or multiple address capability. The address bits (nnnnn) can range from 1 to 31, depending on the received AES message. Values of 1 to 30 are also called AES and value 31 is IES.
- NMP - Next multiple primary. Auto address group, ready class, 101 111nnnnn. This mnemonic represents the incremented primary address message that multiple address devices send on to the next device on the loop. The lower five bits can range from 1 to 31 depending on the value of the received AMP message. The address value 31 is also known aso IMP and value less than 31 are also called AMP.
- NOP - No Operation. Universal command group, command class, 100 00010000. This is the universal no operation command. It is useful for disabling asynchronous request mode since any universal command disables it and the NOP command has no other effects.

- NRD - Not ready for data. Addressed ready group, ready class, 101 01000010. When the controller or a device enabled by the controller needs to interrupt an active talker during a data transmission, it does so by holding the next data byte and replacing it with the NRD message. This message signals the talker that it should terminate its transmission. When the NRD returns to the sourcing device, it will then send the held data message. When this data message is received by the talker, it sends the EOT message to the controller. If an active talker is directed to continue the data transfer (with a send data message), it must continue at the point of interruption unless directed otherwise in a device dependent manner defined for that purpose.
- NRE - Not remote enable. Universal command group, command class, 100 10010011. With this command, the controller causes all devices to be placed under local control; that is, they will respond to their front panel controls and not to programming information received from the loop. Devices which do not implement the remote local interface function will simply ignore this command.
- NUL - Null command. Addressed command group, command class, 100 00000000. This is the addressed no operation command. Devices do not perform any action in response to this command so it is useful for such operations as testing the loop handshaking.
- OSA - Other secondary address. Secondary address group, command class, 100 011ttttt. This mnemonic represents any secondary address command whose address bits (ttttt) do not match the address assigned to this particular device. It causes talker functions to become unaddressed.
- OTA - Other talk address. Talk address group, command class, 100 010ttttt. This mnemonic represents any talk address command that contains an address (ttttt) that does not match the address assigned to this particular device. Since there can only be one active talker on the loop at a time, this device will return its talker function to the idle state.
- PPD - Parallel poll disable. Addressed command group, command class, 100 00000101. This command is used by the controller to cause the devices which are listen addressed to no longer respond to parallel polls. The parallel poll function returns to its idle state.

- PPEn - Parallel poll enable n. Addressed command group, command class, 100 1000sbbb. This command allows the controller to configure devices that are addressed to listen to respond to parallel polls in various ways. The s bit indicates the sense of the device's response. If s is 1 the device will set the assigned bit of an IDY frame if it needs service, if s is 0 the device will set the assigned bit if it does not need service. The lower three bits (bbb) indicate the binary bit number on which the device must respond. 000 indicates bit D0, 001 indicates D1, ..., 111 indicates D7.
- PPU - Parallel poll unconfigure. Universal command group, command class, 100 00010101. The controller uses this command to disable all devices on the loop from responding to parallel polls. The parallel poll function in each device will return to its idle state.
- RDY - Ready. 101 xxxxxxxx. This major class of messages is used for several different purposes, including device handshake functions and address configuration. Most are sourced by the active controller, but some may be sourced by other devices under certain conditions.
- REN - Remote enable. Universal command group, command class, 100 10010010. With this command, the controller enables the loop for remote operation. When a device is addressed to listen, it enters its remote control state and will no longer respond to its front panel controls. Devices which do not implement this function will simply retransmit this command and take no other action.
- RFC - Read for command. Ready class, 101 00000000. The controller uses this ready frame as the handshake after each command so that it knows that all devices on the loop have received the command and are ready to receive the next. Every command must be immediately followed by the RFC message.
- SADn - Secondary address n command. Secondary address group, command class, 100 011aaaaa. This command provides the secondary address to enable talkers and listeners which respond to two byte addresses. The lower five bits represent the binary address n, which can range from 0 to 30. 31 is an illegal address. The secondary address must follow the talk or listen address command immediately to enable the device to talk or listen.

- SAG - Secondary address group. Command class, 100 011xxxxx. This group of commands contains the secondary addresses, that is, SAD commands. These are only used for devices which respond to extended or multiple address modes.
- SAI - Send accessory identification. Addressed ready group, ready class, 101 01100011. This message is used by the controller to cause the addressed talker to begin sending its accessory ID byte(s). The talker replaces the SAI with the accessory ID on the loop, and terminates with the proper EOT message, just as in a data transmission. If the device does not have accessory ID capability, it merely sends the SAI back to the controller to indicate that the device cannot respond. Accessory ID consists of a single byte whose high order four bits indicate the device class (e.g. printer, mass storage, etc.) and low order four bits represent the device type.
- SDA - Send data. Addressed ready group, ready class, 101 01100000. This message is used by the controller to direct the addressed talker to begin sending its data. The talker replaces the SDA with its first byte of data and continues to send data until no more is available. The talker follows its last data byte with the proper EOT message. If the device cannot source data, it merely returns the SDA to the controller. If the talker has no data ready, it sends ETO.
- SDC - Selected device clear. Addressed command group, command class, 100 00000100. The SDC command causes active listeners to perform their device dependent clear function. SDC has no affect on other interface functions.
- SDI - Send device ID. Addressed ready group, ready class, 101 01100010. SDI is used by the controller to cause the talker addressed device to begin sending its device ID string. The talker replaces the SDI with its ID string on the loop and terminates the transmission with the proper EOT message. The ID string consists of ASCII characters followed by carriage return and linefeed. Typically, the ID has the following form: two characters for the manufacturer code, up to five characters for the model number, one model number revision character, and optionally other information that describes options or other information that the designer feels should be included to clarify the identification and capabilities of this device. If a device does not implement device ID, the SDI message will simply be returned to the controller.

- SOT - Start of transmission. Addressed ready group, ready class, 101 01100xxx. This subgroup of messages includes the SDA, SST, SDI, SAI, and TCT messages. They all serve to enable the beginning of transmission from a device other than the controller (except for TCT, which enables the new controller). The other device will source the proper EOT message when finished to signal the controller to take over once again. In the case of TCT (take control) the new controller remains in control of the loop and does not source the EOT. This group of messages does not go completely around the loop, but is replaced by the first message from the enabled device.
- SRQ - Service request. Data or end class or identify class, 0x1 xxxxxxxx or 111 xxxxxxxx. The SRQ bit is bit C0 of data, end, and identify messages. Devices may set this bit when they have a need for service from the controller. The bit then represents the logical OR of the various devices' individual service bits. The controller will generally need to perform a serial poll operation to find out which device needs service. The command and ready classes of messages do not have a service request bit and, therefore, do not transmit this message.
- SST - Send status. Addressed ready group, ready class, 101 01100001. The controller uses this message to cause the addressed talker to begin sending its status byte(s). The talker replaces the SST with its status information on the loop. When finished, the talker then transmits the proper EOT message. Two bits of the first byte of status are reserved for specific purposes. If bit D7 (msb) is set, then the first byte of status represents a coded system status message. If it is clear, the lower 6 bits (D5-0) are device dependent. Bit D6 of the first status byte is always equal to the device's local message rsv. If the device does not implement serial poll (status), the SST frame is simply returned to the controller.
- TADn - Talk address n command. Talk address group, command class, 100 010aaaaa. The controller uses this command to enable one device on the loop to be the active talker. The lower five bits (aaaaa) represent the device's address (primary address in the case of those devices which respond to a two byte address). Address values can range from 0 to 30. TAD with address 31 is the UNT (untalk) command. If an addressed talker receives UNT or a TAD with an address that does not match its own, it must become untalked.

- TAG - Talk address group. Command class, 100 010xxxxx. This group includes all primary talk address commands and the untalk command. Secondary addresses are contained in the SAG group.
- TCT - Take control. Addressed ready group, ready class, 101 01100100. The active controller uses this message to pass control of the loop to another controller. The device to which control is passed must first be talk addressed, then the current controller sends the TCT message. Upon receipt of the TCT, the talker addressed device becomes the active controller, and replaces the TCT with its first interface message. If the device cannot accept control of the loop, it merely retransmits the TCT which returns to the current controller, which resumes active control of the loop.
- UCG - Universal command group. Command class, 100 x001xxxx. This group of commands includes all those to which all devices respond whether they are currently addressed or not.
- UNL - Unlisten. Listen address group, command class, 100 00111111. This command causes all addressed listeners to return to the idle state. The controller normally will use this command to reset the listeners before addressing a new listener(s) for the next data transmission.
- UNT - Untalk. Talk address group, command class, 100 01011111. This command causes the addressed talker to return to the idle state. Since a new talk address causes the previous talker to become unaddressed anyway, this command is only useful in certain special cases, such as when it is necessary to have no talker addressed device on the loop.

ZES - Zero extended secondary. Auto address group, ready class, 101 11000000. This message is used by the controller to assign secondary addresses to those devices which have multiple address capability. After each device has received its primary address via the AMP frame, it waits to recognize the ZES frame. When it is received, the low order five bits are incremented by the number of addresses reserved for this device and the frame is then sent back to the controller which now knows how many addresses there are in that device. Since the device will only respond once to the ZES message, the controller sends it once for each device in turn. After this, all devices are configured and can respond normally to their assigned primary and secondary addresses.

Appendix C. Message Coding

C.1 Command Coding

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF	
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
0x 0000	NUL	GTL			SDC	PPD			GET							ELN	ACG
1x 0001	NOP	LLO			DCL	PPU			EAR								UCG
2x 0010	LAD 0	LAD 1	LAD 2	LAD 3	LAD 4	LAD 5	LAD 6	LAD 7	LAD 8	LAD 9	LAD 10	LAD 11	LAD 12	LAD 13	LAD 14	LAD 15	LAG
3x 0011	LAD 16	LAD 17	LAD 18	LAD 19	LAD 20	LAD 21	LAD 22	LAD 23	LAD 24	LAD 25	LAD 26	LAD 27	LAD 28	LAD 29	LAD 30	UNL	LAG
4x 0100	TAD 0	TAD 1	TAD 2	TAD 3	TAD 4	TAD 5	TAD 6	TAD 7	TAD 8	TAD 9	TAD 10	TAD 11	TAD 12	TAD 13	TAD 14	TAD 15	TAG
5x 0101	TAD 16	TAD 17	TAD 18	TAD 19	TAD 20	TAD 21	TAD 22	TAD 23	TAD 24	TAD 25	TAD 26	TAD 27	TAD 28	TAD 29	TAD 30	UNT	TAG
6x 0110	SAD 0	SAD 1	SAD 2	SAD 3	SAD 4	SAD 5	SAD 6	SAD 7	SAD 8	SAD 9	SAD 10	SAD 11	SAD 12	SAD 13	SAD 14	SAD 15	SAG
7x 0111	SAD 16	SAD 17	SAD 18	SAD 19	SAD 20	SAD 21	SAD 22	SAD 23	SAD 24	SAD 25	SAD 26	SAD 27	SAD 28	SAD 29	SAD 30		SAG
8x 1000	PPEO 0	PPEO 1	PPEO 2	PPEO 3	PPEO 4	PPEO 5	PPEO 6	PPEO 7	PPE1 0	PPE1 1	PPE1 2	PPE1 3	PPE1 4	PPE1 5	PPE1 6	PPE1 7	ACG
9x 1001	IFC		REN	NRE							AAU	LPD					UCG
Ax 1010	DDL 0	DDL 1	DDL 2	DDL 3	DDL 4	DDL 5	DDL 6	DDL 7	DDL 8	DDL 9	DDL 10	DDL 11	DDL 12	DDL 13	DDL 14	DDL 15	ACG
Bx 1011	DDL 16	DDL 17	DDL 18	DDL 19	DDL 20	DDL 21	DDL 22	DDL 23	DDL 24	DDL 25	DDL 26	DDL 27	DDL 28	DDL 29	DDL 30	DDL 31	ACG
Cx 1100	DDT 0	DDT 1	DDT 2	DDT 3	DDT 4	DDT 5	DDT 6	DDT 7	DDT 8	DDT 9	DDT 10	DDT 11	DDT 12	DDT 13	DDT 14	DDT 15	ACG
Dx 1101	DDT 15	DDT 16	DDT 18	DDT 19	DDT 20	DDT 21	DDT 22	DDT 23	DDT 24	DDT 25	DDT 26	DDT 27	DDT 28	DDT 29	DDT 30	DDT 31	ACG
Ex 1110																	
Fx 1111																	

C.2 Ready Coding

	x0 0000	x1 0001	x2 0010	x3 0011	x4 0100	x5 0101	x6 0110	x7 0111	x8 1000	x9 1001	xA 1010	xB 1011	xC 1100	xD 1101	xE 1110	xF 1111	
0x 0000	RFC																
1x 0001																	
2x 0010																	
3x 0011																	
4x 0100	ETO	ETE	NRD														ARG
5x 0101																	ARG
6x 0110	SDA	SST	SDI	SAI	TCT												ARG
7x 0111																	ARG
8x 1000	AAD 0	AAD 1	AAD 2	AAD 3	AAD 4	AAD 5	AAD 6	AAD 7	AAD 8	AAD 9	AAD 10	AAD 11	AAD 12	AAD 13	AAD 14	AAD 15	AAG
9x 1001	AAD 16	AAD 17	AAD 18	AAD 19	AAD 20	AAD 21	AAD 22	AAD 23	AAD 24	AAD 25	AAD 26	AAD 27	AAD 28	AAD 29	AAD 30	IAA	AAG
Ax 1010	AEP 0	AEP 1	AEP 2	AEP 3	AEP 4	AEP 5	AEP 6	AEP 7	AEP 8	AEP 9	AEP 10	AEP 11	AEP 12	AEP 13	AEP 14	AEP 15	AAG
Bx 1011	AEP 16	AEP 17	AEP 18	AEP 19	AEP 20	AEP 21	AEP 22	AEP 23	AEP 24	AEP 25	AEP 26	AEP 27	AEP 28	AEP 29	AEP 30	IEP	AAG
Cx 1100	AES 0	AES 1	AES 2	AES 3	AES 4	AES 5	AES 6	AES 7	AES 8	AES 9	AES 10	AES 11	AES 12	AES 13	AES 14	AES 15	AAG
Dx 1101	AES 16	AES 17	AES 18	AES 19	AES 20	AES 21	AES 22	AES 23	AES 24	AES 25	AES 26	AES 27	AES 28	AES 29	AES 30	IES	AAG
Ex 1110	AMP 0	AMP 1	AMP 2	AMP 3	AMP 4	AMP 5	AMP 6	AMP 7	AMP 8	AMP 9	AMP 10	AMP 11	AMP 12	AMP 13	AMP 14	AMP 15	AAG
Fx 1111	AMP 16	AMP 17	AMP 18	AMP 19	AMP 20	AMP 21	AMP 22	AMP 23	AMP 24	AMP 25	AMP 26	AMP 27	AMP 28	AMP 29	AMP 30	IMP	AAG

C.3 Frame Hierarchy

```

DOE . . . . DAB
      . DAB(SRQ)
      . END
      . END(SRQ)

IDY . . . . IDY
      . IDY(SRQ)

RDY . . . . RFC
      .
      . ARG . . . . EOT . . . . ETO
      .           .           . ETE
      .           .           .
      .           . NRD
      .           .
      .           . SOT . . . . SDA
      .           .           . SST
      .           .           . SDI
      .           .           . SAI
      .           .           . TCT
      .
      .
      . AAG . . . . AAD(0-30)
      .           . NAA(1-31)
      .           . IAA
      .           .
      .           . AEP(0-30)
      .           . IEP
      .           .
      .           . ZES
      .           . AES(0-30)
      .           . NES(1-31)
      .           . IES
      .           .
      .           . AMP(0-31)
      .           . NMP(1-31)
      .           . IMP

CMD . . . . ACG . . . . NUL
      .           . GTL
      .           . SDC
      .           . PPD
      .           . GET
      .           . ELN
      .           . PPE0(0-7)
      .           . PPE1(0-7)
      .           . DDL(0-31)
      .           . DDT(0-31)
      .
      . UCG . . . . NOP
      .           . LLO
      .           . DCL
      .           . PPU
      .           . EAR
      .           . IFC
      .           . REN
      .           . NRE
      .           . AAU
      .           . LPD
      .
      . LAG . . . . LAD(0-30)
      .           . MLA(0-30)
      .           . UNL
      .
      . TAG . . . . TAD(0-30)
      .           . MTA(0-30)
      .           . OTA(0-30)
      .           . UNT
      .
      . SAG . . . . SAD(0-30)
      .           . MSA(0-30)
      .           . OSA(0-30)

```

C.4 Accessory Identification

Accessory ID provides HP-IL Controllers with the ability to quickly identify the devices on the loop according to device functions. The accessory ID consists of a four bit class descriptor, and a four bit type field. The class descriptor indicates what main function the device provides such as printer, mass storage, etc. The type field indicates specific attributes about the device. If a device exhibits similar characteristics as other devices within a particular class and type, the device should respond to accessory ID with that class and type. If no type exists within a device's class that closely matches its attributes, the device should respond with type E of that class. The extended class (Fx) and extended type (xF) are reserved to allow new classes and types.

It is very strongly recommended that all future devices designed for HP-IL systems respond to accessory ID. HP-IL controllers should be able to expect the accessory ID to be available so that certain system I/O functions can be executed automatically.

Classes and Types Defined (hex):

0x Controllers

- 00 Limited controller capability mostly automatic system I/O functions.
example device: HP-41C
- 01 Full instrumentation controller; completely manual (programatic) control.
example device: HP Series 80
- 02 Full interface controller completely automatic including control passing during I/O operations
- 03 Full interface controller partially automatic
- 0E general controller
- 0F extended (reserved)

1x Mass storage devices

10 Seek/read/write protocol using device dependent commands
as defined by the HP 82161A Digital Cassette Drive.

example device: HP 82161A Digital Cassette Drive

1E general mass storage

1F extended (reserved)

2x Printers

20 24 column; HP escape sequences; column dot graphics
example device: HP 82162A Thermal Printer

21 80 column; HP escape sequences; column dot graphics
example device: HP 82905B Impact Printer

22 80 column; HP escape sequences with no graphics
example device: HP 2671A Thermal Printer

23 80 column; HP escape sequences; HP raster graphics
example device: HP 2671G Thermal Printer

24 80 column; HP escape sequences; HP raster graphics
example device: HP 2673A Thermal Printer

2E general printer

2F extended (reserved)

3x Displays

30 32 column; HP escape sequences; no graphics
example device: HP 82163A Video Interface

3E general display

3F extended (reserved)

4x Interfaces

40 HP-IL/GPIO interface
example devices: HP 82165A, HP 82166A

41 HP-IL modem
example device: HP 82168A

42 HP-IL/RS-232-C interface
example device: HP 82164A

43 HP-IL/HP-IB interface
example device: HP 82169A

4E general interface

4F extended (reserved)

5x Electronic Instrumentation.

51-57 If bit D3 is 0 the lower three bits (D2-D0) represent the functions contained in this device. D2, D1 and D0 are defined as signal source, signal switch and signal measurement functions respectively. For example a device that can act as both a signal source and measurement device would respond 55.

5E general electronic instrument

5F extended (reserved)

6x Graphic I/O

60 HP-GL compatible
example device: HP 7470A Plotter

6E general graphic I/O device

6F extended (reserved)

7x Analytical and Scientific Instrumentation

7E general analytical or scientific instrument

7F extended (reserved)

Ex General devices. This class consists of devices that do not easily fit into the other classes listed.

E0 EPROM programmer

EE general

EF extended (reserved)

Fx Extended class (usage not currently defined)

C.5 System Status Messages

HP-IL has the capability for a device to report status to the controller in an interface defined code called system status. It is required that all devices on HP-IL reserve the most significant bit of the first byte of status to indicate system status. Bit 7 (msb) is required to be zero when the device sources device dependent status, and one when the device sources system status. More than one byte of status may be sent, but all bytes after the first one are designer specified.

It is very strongly recommended that devices use system status messages whenever possible. This allows generalized controllers to be implemented that can understand status messages from any device on the loop. Most devices can describe all possible status with the system messages defined. If more specific information is necessary or desirable, the device may send additional bytes of status.

Generally, there two types of status messages. State messages represent the overall status of the device. A state message will remain true within a device until another message of higher priority replaces it. All OK is an example of a state status message. Event messages indicate an occurrence within a device that does not necessarily affect the current state of the device. Once the controller has been informed of the situation through a serial poll, the device removes the event system status message. Data Error is an example of an event status message.

Except for the All OK message, the event and state system status messages can be distinguished by the value of bit D5. If bit D5 is set, the message is a state or condition. If bit D5 is clear, the message is an event. The all OK message is a state, but the value of this message is all zeros.

Devices must select the most important system status message to report during each serial poll. The most important status message is always the one having the highest priority that is currently true within the device. The following table lists all system status messages and their meanings in priority order with the highest priority messages appearing first.

System Status Messages

Device Events

DDDDDD

543210 Message definition

- 000110 Self test failure. The device has discovered a condition that makes proper operation impossible to guarantee. Highest priority of all system status messages.
- 001000 Powering Down. The device indicates to the controller with this message that it is about to power down and therefore, the loop will become inoperative shortly. The message is only useful if the device can delay powering down long enough to report the condition. Note that this message is not to be used if the device is powering down in response to an LPD command.
- 001001 External Service Request. With this message, the device indicates to the controller that an external input (such as a switch) has been activated.
- 000010 Manual intervention required. The device cannot function as designed until an action has been performed by the operator.
- 000011 Data error. The device has detected a situation that caused information to be lost in a device dependent manner not associated with the talker function. If data is lost due to a loop transmission error, the End Of Transmission with Error (ETE) ready frame will be sourced by the talker.
- 000111 Command Error. This message indicates to the controller that an invalid device command was received.
- 000101 No room. The device does not have sufficient space for the data being received from the loop and the situation will not be resolved without intervention.
- 000100 Device error. The device is in an invalid state or cannot perform the desired operation. This status may be the result of an incorrect sequence or combination of device commands or may be due to an error condition within the device.

System Status Messages (continued)

- 000001 Low battery. The device has detected that the battery power has reached a critical level. Future performance of the loop is endangered if this message is ignored.
- 001010 Device Dependent Service Request. The device has detected an event which requires attention from the controller. This message should be used only if no other system status message is appropriate.
- 011111 ASCII Follows. The bytes following this message (until an ETO) represent an ASCII message for the user or operator. Lowest priority of device event messages.

Device States

DDDDDD

543210 Message definition

- 100000 Request control of loop. A device may indicate to the current controller with this message that it has need for control of the loop. Highest priority of device state messages (lower priority than event messages).
- 100010 Ready to send data. The device has data available and is currently ready to source it on the loop.
- 100001 Ready to receive data. The device indicates with this message that it is currently ready to accept data from the loop.
- 100011 Not ready to receive or send data. The device indicates to the Controller with this message that it has no data ready to send and that it is not able or ready to accept data. Without any intervention, the device will eventually correct the situation and become ready.
- 000000 All OK. This message indicates that the device has no need for attention and is currently ready to perform the operations for which it was designed. Lowest priority of all system status messages.



HEWLETT
PACKARD

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