ElectronicDesign®

FOR ENGINEERS AND ENGINEERING MANAGERS—WORLDWIDE

DECEMBER 24, 1981

Special Report:
Logic arrays turn to CMOS for speed, density

ISSCC '82:
An advance look at the coming big chips

Flash converter builds superfast 8-bit systems

Bit-serial interface loop links portable instrumentation

Automation arrives for bench and field instrumentation

Out of the engineering laboratories of two major divisions of one of the world's largest instrument companies, Hewlett-Packard, comes HP-IL—a new concept in interfacing low-cost instrumentation on a serial loop. Like its predecessor, HP-IB, which went on to become an important universal interface standard for sophisticated laboratory and production equipment, HP-IL promises to open new doors in testing and measurement. Plainly put, for lower-cost, lower-power, or portable equipment—like hand-held computers and small DMMs—there is nothing else.

On the pages that follow, the men heading up the two divisions share their assessments of HP-IL and talk about its significance. Then a technical article unwraps the essential details a designer will need to know to take full advantage of HP-IL. It is followed by a second article that shows how a remarkable new DMM uses the loop to its best advantage.

The division heads

William G. Parzybok *is now general manager of Hewlett-Packard's recently formed Electronics Measurement Group. Previously, he was the general manager of the Loveland Instrument Division, which manufactures a wide range of test and measurement equipment.*

With the launching of HP-IL—the Hewlett-Packard Interface Loop—for linking low-cost, low-power instrumentation—automation has finally come to the designer's bench and to the field. The HP-IL will not only let instruments and hand-held computers work together, it will also tie in computer peripherals: small cassette drives, printer/plotters, and the like.

Automation is something we don't normally associate with a designer's bench or with portable, battery-powered equipment. But it stimulates the imagination.

We could picture a designer setting up a small experiment, with the HP-IL and HP-41 hand-held computer running a voltmeter, and maybe a printer/plotter or a tape cassette with programs on it—and perhaps the experiment runs itself overnight.

This is but one example of applications that might not have been thought of in conjunction with automation—opening those doors is the major significance of the HP-IL.

Of course, having a low-cost standard interface means that new products will be able to work on the loop without special or custom circuitry. More importantly, HP-IL fights obsolescence. That is, when a small computer is surpassed by a new model, its peripherals won't become obsolete, or vice versa. The same is true of instrumentation with HP-IL capability. Recall that with our other standard interface—the HP-IB—newer computers can connect to any HP-IB equipment.

It's important to stress that we are in no way replacing the HP-IB. HP-IL satisfies an entirely different set of requirements, for low-cost, low-power equipment. The HP-IB does not lend itself to that. Instead, it excels where higher speeds and measuring sophistication are needed—an avionics test system, for example.

We do not plan to promote HP-IL as an industry standard. Certainly, if HP-IL catches on and everybody likes it, we'll be glad to help in making it a standard, much in the same way we did with IEEE-488. Right now, though, HP-IL is a company standard and we are planning to develop our own products for it.

We can't say exactly what kinds of HP-IL instruments and computers will appear—other than the ones we are introducing now—but it will be those classes that lend themselves to low-cost automation.

The evolutionary trend to low-cost, interfaceable products will continue, and we are going to see more HP-IL products—from HP and perhaps from others. It all adds up to a good deal for the entire industry.

For good examples of the pacesetters, look at our new 3468A multimeter and the 41 hand-held computer. The 41 puts a tremendous amount of computer power in your hand. The multimeter, if I may say so, is pretty fabulous too: 5-digits, no pots, no trimmers, automatic calibration, built-in smarts, automation—all for $695. With its built-in HP-IL, there's even wider appeal.

Yet there's more to the story. We have designed the manufacturing process along with our new digital multimeters. We are automating the assembly process, keeping the tightest controls ever. Our goal is defect-free design, parts, assemblies and processes.

Our goal is that these products will be the most reliable DMMs ever introduced.

talk about HP-IL

Fred Hanson *is the manager of Hewlett-Packard's Corvallis, OR division, which makes hand-held computers and associated peripheral equipment.*

As I look at HP-IL, I see an interface directed toward very low-cost and highly portable products. Just as the HP-IB brought a system capability within reach of people building modest-sized semipermanent and semiportable installations, so HP-IL will continue the process, bringing a system capability to low-cost, truly portable applications.

Of course, you have to give up something when you slash costs dramatically, and in the case of HP-IL, what you give up is speed. Thanks to recent advances in high-density integrated circuitry, you really don't give up anything else; in fact HP-IL has a few features that the faster HP-IB lacks.

Addressing the speed question, it's important to note that HP-IL uses a serial transmission format, whereas the HP-IB has a byte-wide bus structure. Also, the HP-IB uses fast bipolar circuitry, but in the interests of minimizing power consumption, HP-IL employs CMOS, which, of course, further limits its speed. The end result is that the HP-IL has a theoretical maximum transmission rate of 20 kbytes per second. However, currently available components can support about 5 kbytes/s.

But let's look at the advantages that the HP-IL brings to small, portable systems. First of all, it is a low-power interface, which means that it's suitable for battery-powered applications. In this connection, it is worth pointing out a special HP-IL feature not found on the HP-IB —remote power-down control. This feature allows the controller to put the instruments in a low-power, standby state—which is important for managing power in battery-operated systems.

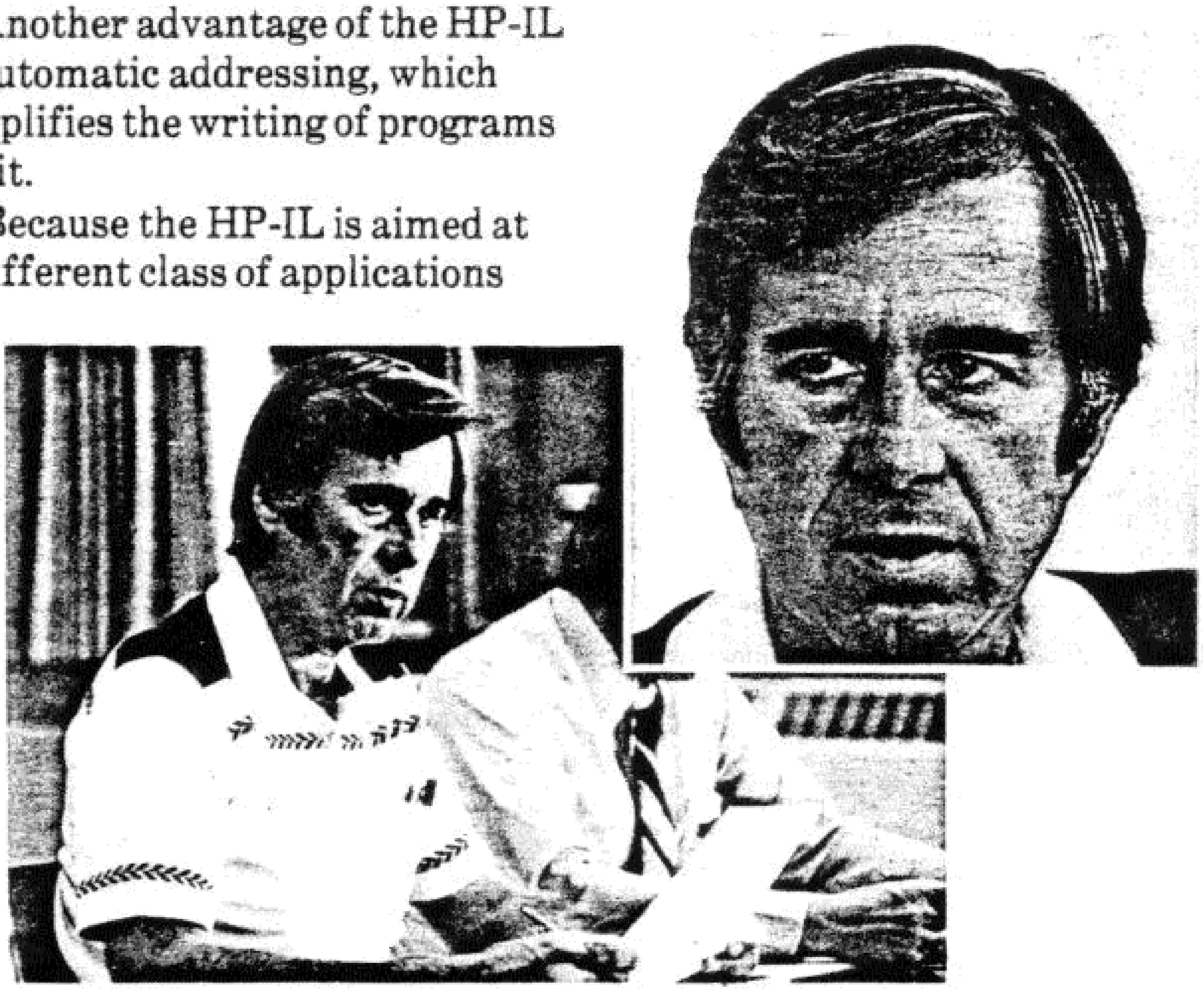
The loop's low power consumption comes not only from its CMOS circuitry and serial data format, but also from its architecture. Each device on the loop retransmits every message frame that it receives. Thus, it needs to provide only enough power to reach the next device down the line. A two-wire differential-drive link provides good noise immunity and allows line lengths of up to 100 meters, compared with 20 meters for the HP-IB. Of course, the two wire connectors used with the loop are much smaller than the 25-pin units used with the bus—an important consideration when dealing with instruments and computers small enough to fit in your hand.

Another advantage of the HP-IL is automatic addressing, which simplifies the writing of programs for it.

Because the HP-IL is aimed at a different class of applications than the HP-IB, it will also work with a different class of instruments. A typical HP-IL instrument will be small, portable, inexpensive, and fairly simple. It will, typically, be unable to interface with the HP-IB because that would significantly increase its price and power consumption, and also its size.

Whereas the word "system" until now conjured up an image of several fairly large boxes connected by some substantial cabling to a desktop computer, microcomputer or a minicomputer, I can see an HP-IL system made up of nothing more than a HP 41 computer, a DMM, and a printer. Such a system could be hooked up in a matter of minutes and might be used on a project that only lasts a day.

Imagine a complete system that fits into a briefcase for remote and portable applications. We're not quite there yet, but I expect that we will be soon.



Interface system weds instruments to small computers

The HP-IL links hand-held and bench-top instruments, computers, and peripherals in a serial loop that can run off batteries.

Quick to take their place in the field and at the engineer's bench, small instruments and even computers—down to hand-held, calculatorlike versions—have stuck to relatively obvious accomplishments. The missing rung on their ladder to true stardom has been a standard interface system appropriate to their low price, low power, and small size.

The Hewlett-Packard Interface Loop is about to change all that. Linking up to 31 devices—with extended addressing, a whopping 961 pieces of equipment—via simple and inexpensive two-wire cabling, and capable of running off batteries, the HP-IL is a low-cost, portable interface system. Clearly, its balance of speed vs power dissipation (not to mention cost) puts it into a separate class geared for portable computers (Fig. 1). What's more, it throws into the bargain a communications scheme that all but eliminates errors.

A simple loop architecture

The HP-IL permits various devices to communicate with one another within a loop architecture. Communication is unidirectional in a bit-serial manner. Each device has one input and one output port by which it hooks up to the loop in a daisy chain. A noncompatible device with a general-purpose input/output (GPIO) port can interface with the HP-IL through the HP82166A HP-IL converter (see Fig. 2 and "Making General-Purpose I/O Devices Compatible with the HP-IL"). Each device retransmits information on the loop after receiving it from the preceding device. Each device also supplies enough power to drive the next one, thus equalizing and minimizing the power requirements of each of the loop's participants.

Steve Harper, Technical Staff Member
Carl Landsness, Technical Staff Member
Roger Quick, Project Manager
Hewlett-Packard Co., Corvallis Division
1000 N.E. Circle Blvd., Corvallis, OR 97330

The loop architecture employs simple, inexpensive two-wire cabling, such as zip cord, instead of the more expensive shielded cables generally required for other interface systems. The two-wire differential-drive medium is sufficiently immune to electromagnetic interference for the intended applications.

Data propagated within the loop makes a complete circuit, along with the associated protocols and commands. Thus, after all other devices on the loop have received and retransmitted it, information returns to the device that originated it. All errors can then be detected, since the originating device can compare the transmitted with the received data for accuracy.

A master-slave concept

The HP-IL employs a master-slave architecture, wherein a talker device on the loop is designated as the loop controller and master, and all other loop devices, talkers or listeners, are slaves. The controller transmits all HP-IL system commands to all other devices on the loop.

A device may be capable of performing only one or two of the three loop roles—listener, talker, controller. (Of course, insofar as each device must pass along every message, each both "listens" and "talks," but that loose sense is not what is meant when speaking of "listeners" and "talkers.")

In general, every HP-IL system has a controller, but under certain circumstances a small, simple system can be constructed with only a talker and one or more listeners. In such a case, each device must operate either only as a talker or only as a listener. One example is a voltmeter (talker) logging readings on a printer (listener).

Larger systems contain several talker and listener devices. In that case, the loop's controller permits only one talker at a time to be active. On the other hand, several listeners may be enabled simultaneously, should that be desired.

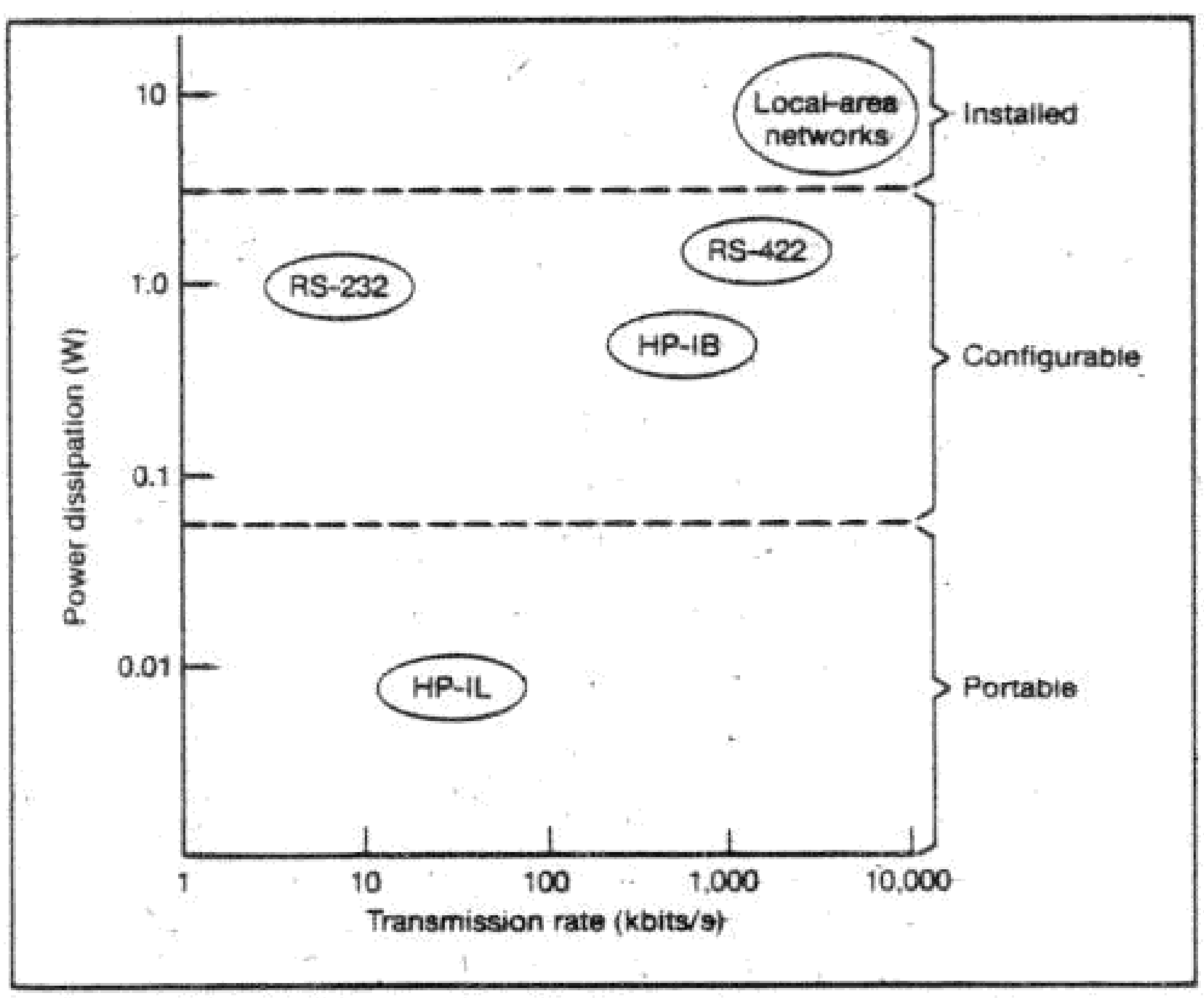
It is even possible to have several controllers on the HP-IL. One controller is assigned to take charge when the system is powered on, and there are protocols to allow others to take turns. However, only one controller may be active at a time. Should a controller with a higher-priority task be held up by a long-winded talker, one of the protocols interrupts the talker's conversation in an orderly fashion.

Examples of suitable controller devices are the HP-41 hand-held computer and the HP-85A personal

computer (see "Interfacing the HP-41 with the HP-IL"). The HP82162A thermal printer is one example of a listener (see "A Thermal Printer/Plotter for the HP-IL"). A device with both talker and listener capabilities is the HP82161A digital cassette drive (see "A Cassette Drive for the HP-IL"), as is the HP3468A digital multimeter (see the following article).

As previously mentioned, the HP-IL design was driven by the need to be compatible with small, low-cost, low-power devices. The result is a careful blend of serial architecture, balanced differential drive, a three-level encoding scheme, and the use of CMOS LSI circuits and pulse transformers, all of which are highly interdependent.

The system's two-wire transmission, plus the need for asynchronous operation, dictates the use of a self-clocking code. Most codes in this class employ two signal levels and encode their logical information as a function of time (nonreturn to zero inverted, F2F, Manchester, Delta distance, for instance), so that some form of pulse-width timing is always required. Although these codes are fairly insensitive to timing variations, they have their limits. These limitations are nonexistent, however, with three-level code, in which information is a function of level transitions only (Fig. 3).



1. When fast data transmission is not an overriding factor, the Hewlett-Packard Interface Loop, HP-IL, offers the best cost/power-dissipation/speed/size tradeoffs of all major interfaces. The loop system is highly flexible for field applications, owing to its portability and the possibility it offers of battery operation.

The use of a three-level code produces numerous system design and operational advantages. For one, every message bit is totally asynchronous with respect to adjacent bits; no "preamble," or start bits, is required to establish timing reference, as is the case for most two-level type codes. Furthermore, the three-level code lends itself nicely to simple detection schemes. For another, two-level codes with narrow pulse widths (say, 1 μs) usually require a reasonably accurate timer (a digital counter or a one-shot), which is not easily achieved with a low-power IC technology. Lastly, but most importantly, the particular code employed is highly insensitive to pulse distortions caused by variable-length cables, receiver load circuits, and variable-speed drivers.

The three-level code used in the HP-IL defines four types of logical bits—1, 0, 1S, and 0S—the nominal voltage levels of which are fairly low to conserve power. The latter two bits are specially encoded versions of 1 and 0, respectively, and are used at the start of each message frame for synchronization. This bit provides an unequivocal start of frame reference.

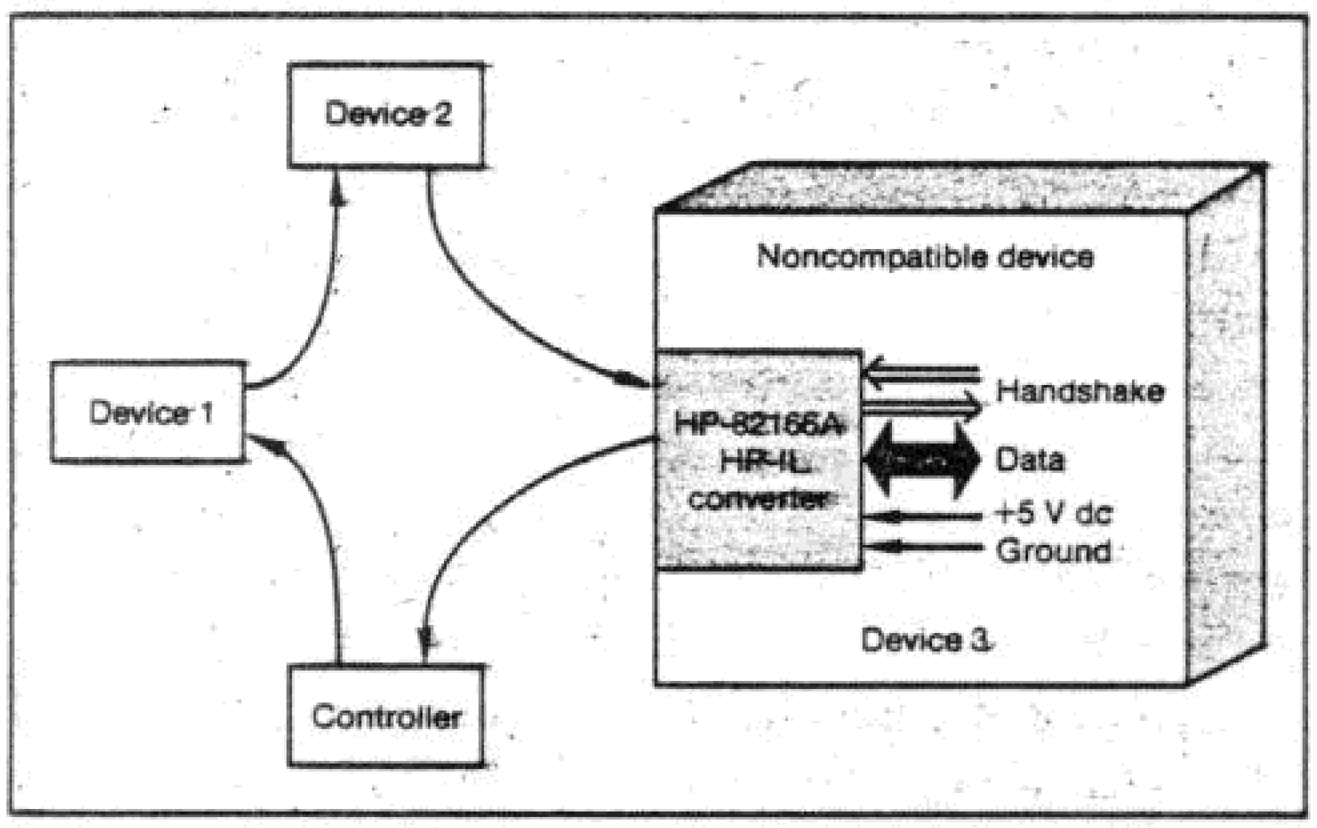
The three-level coding technique trades off a slight amount of bit density for reliability. Here, a high pulse (+1.5 V) followed by a low (−1.5 V) defines a logical 1, and the reverse specifies a logical 0. The nominal pulse width is 1 μs, and each bit sequence is always followed by a minimum delay (0 V) of about 2 μs. Should electrical interference such as a power-line transient cause an isolated high or low pulse to occur, it is ignored unless it is immediately followed by a pulse of the opposite polarity.

Because coding is asynchronous with respect to bit and frame, no common system clock is needed, making it possible for devices operating at different speeds to communicate easily with one another without buffering. Even bits within a frame are asynchronous. For example, a device may retransmit a frame bit as soon as it is received; it need not wait to receive the entire frame.

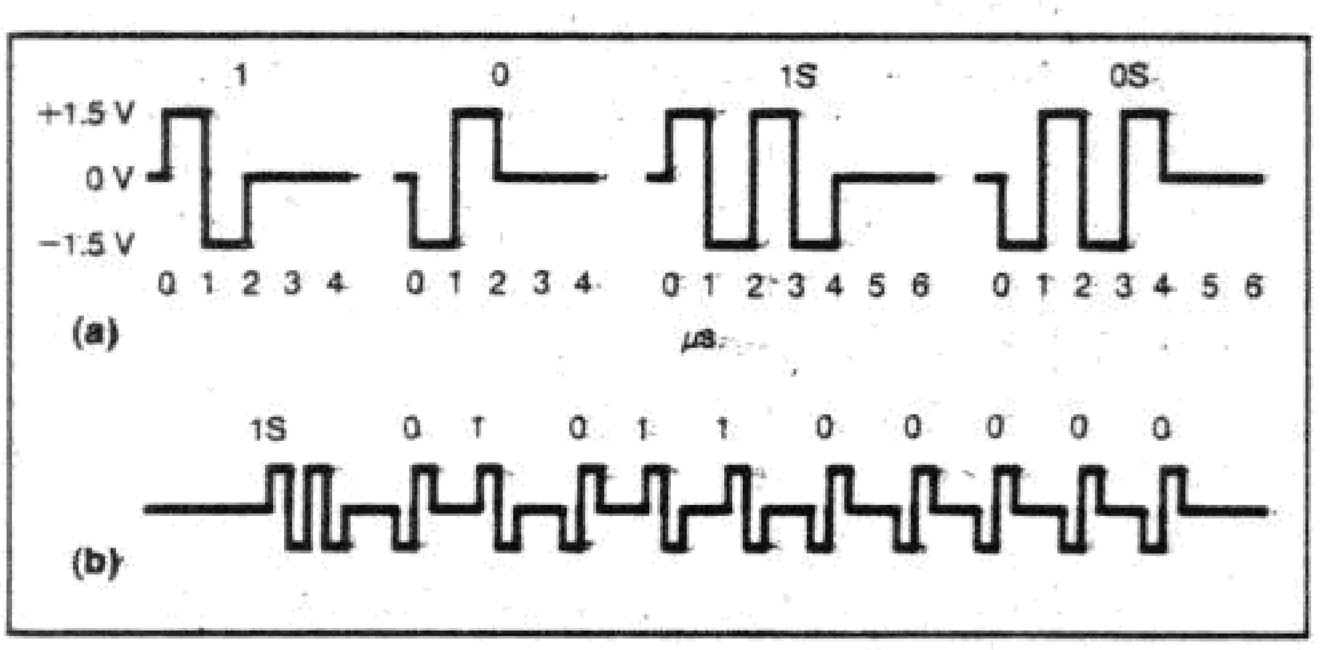
As can be seen from the timing diagrams (Fig. 3), it takes 6 μs to transmit a sync bit and only 4 μs to transmit other frame bits. A complete frame of 11 bits, containing 1 data byte, requires 46 μs for transmission, assuming no additional delay between bits. Assuming, further, no additional delay between frames, the maximum loop data-transmission rate would be over 20 kbytes/s. That is an ideal figure, however, based on theoretical calculations. Actual data-transmission rates vary from 3 to 5 kbytes/s, because of delays due to system hardware and software elements.

Transformers are the key

A key element in the HP-IL is the use of transformer-isolated drivers and receivers. The transformers provide a convenient way of generating the three voltage levels necessary for the code. In fact, a three-level signal as used in the HP-IL is preferable for the transformers because such a scheme has no dc component. Since the transformers are passive, they require no standby power. They also are an efficient means of shifting voltage levels, so that devices with different supply voltages can easily communicate with one another. Furthermore, by stepping a voltage down with near 100% power-conversion efficiency to an HP-IL nominal level of 1.5 V, a transmission line (100 Ω) can be driven with 10 times less power than a system using a 5-V logic level to drive the line. Figure 4 is a simple block diagram of the interface electronics HP-IL implementation.



2. Any noncompatible device with a general-purpose input/output port can be connected to the HP-IL by means of an HP-82166A converter.



3. Timing limitations of self-clocking codes are completely eliminated on the HP-IL with a three-level code (a) which makes information a function of level transitions only. An example of an encoded message—the Send Data ready frame—is shown (b).

The use of transformers also simplifies impedance transformation. The interface requies a nominal Thevenin-equivalent circuit of 1.5 V and 100 Ω for a driver. For a device with a 6-V supply, a 4:1 step-down transformer will produce the required 1.5 V, but the device must provide a 1600-Ω impedance to the transformer. Such a requirement is easily satisfied with MOS IC technology and an external series resistor (for reasonable tolerance), thus eliminating any need for special discrete line drivers.

Since transformers dc-isolate the devices, the latter may operate at different potentials. For example, a digital voltmeter can be used, where its common side may be a few hundred volts above earth ground. DC isolation also totally eliminates any ground loops. Yet another advantage of transformers is that they make possible balanced differential operation, a key element in the HP-IL's ability to operate in noisy environments without earth grounds for shielding.

Finally, without transformers, the interface would require analog-type drivers and receivers, which are difficult to integrate in a power-efficient manner. As it is, very simple digital circuits may be used.

Communicating with an 11-bit frame

A message on the HP-IL consists of one or more 11-bit frames, each of which contains a command or a data byte. A source device sends out a message frame by frame. The destination device (or devices) delays passing on the frame it has received until it is ready for the next one. The source, meanwhile, waits: it will not send out a new frame until it has gotten back the present one.

This hold-until-ready type of handshaking works well for loops with only one destination device; having more than one destination device on the loop, however, slows down data transmission considerably. That is particularly true when a command message is sent by a loop controller to which all devices on the loop must respond.

To speed things up, the HP-IL uses a special command handshake procedure. All devices on the loop pass on a command frame as rapidly as possible while keeping a copy of it. Thus, while the command

message is making the rounds of the loop, some or all of the destination devices are busy executing it in parallel. When the controller receives the message back, it sends out a special frame called "Ready for Command" (RFC), which is not sent on by each destination device until that device is done with the previous command message. This frame is then passed on to the next loop device, and so on, back to the controller, before the controller can send out the next message.

Getting the message

Messages can be divided into two major categories: interface and device. The former usually affect only the interface functions and are needed to configure, address, initiate, and supervise the various HP-IL operations. This class of message, of which the controller's commands are the prime example, is normally regarded as overhead.

Device-dependent messages are made up of data frames. Although these frames are processed by the interface functions, they are directed to the device itself rather than the device interface. Data bytes may represent device programming codes, binary data, or perhaps ASCII characters.

As mentioned previously, each frame consists of 11 bits. The first three (C_3, C_2, and C_1) are control bits that signal the type of message (data, command, etc.); in addition, C_3 also functions as the sync bit, as described earlier. The following eight (D_8 through D_1) indicate the specific message—say, the Unlisten command or the character "A." The Unlisten command frame, for example, would be 100 0011111.

If the first bit, C_3, is 0 (actually, 0S), the frame is a data message. C_2 is an end-of-record flag, and C_1 serves as the service-request bit that devices may

Making general-purpose I/O devices compatible with the HP-IL

The HP82166A HP-IL converter transforms a noncompatible device with general-purpose input/output (GPIO) capabilities into one compatible with the Hewlett-Packard Interface Loop, or HP-IL (see figure). The converter, connected to such a device (for example, a GPIO printer on the loop), allows the loop's controller to interact with the device through it. The noncompatible device thus becomes an HP-IL peripheral.

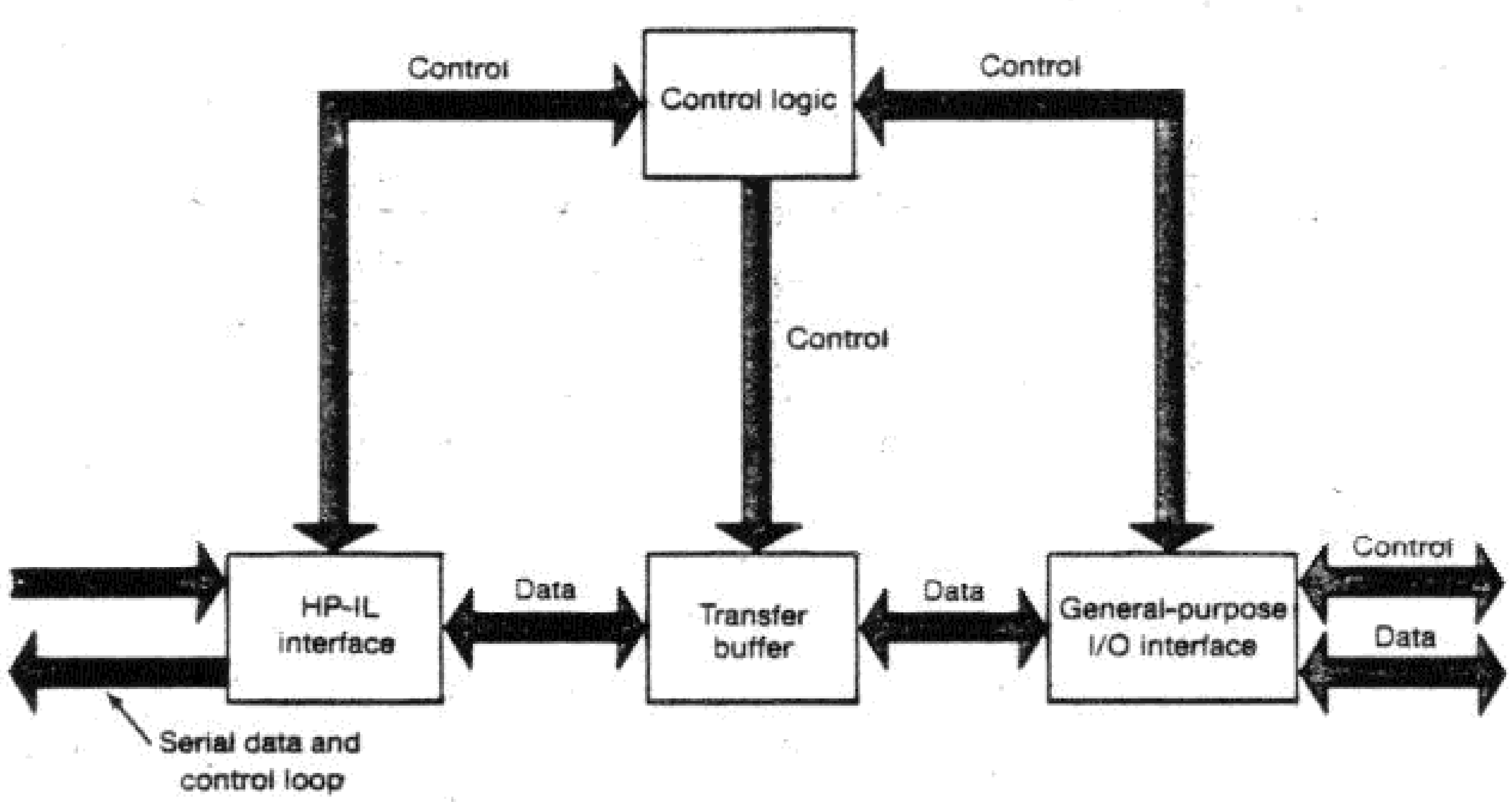
The converter comprises four primary circuits: the HP-IL interface, the general-purpose I/O interface, a transfer buffer, and control logic.

The HP-IL interface portion performs standard operations required by the interface loop, such as maintaining the converter's talker or listener status and accepting and passing HP-IL messages around the loop. The physical connection between the interface and the loop consists of two standard HP-IL receptacles, one for incoming and one for outgoing messages.

The general-purpose I/O portion makes the connection to the device. It includes a physical link consisting of a 34-pin connector on a PC board. The device supplies the converter with power through the connector and sends and receives information over two 8-bit buses.

The transfer buffer consists of 32 registers, each of which can hold an 8-bit byte of information. It stores information sent from the loop to the GPIO device and vice versa, passing it in the order in which it is received (first in, first out).

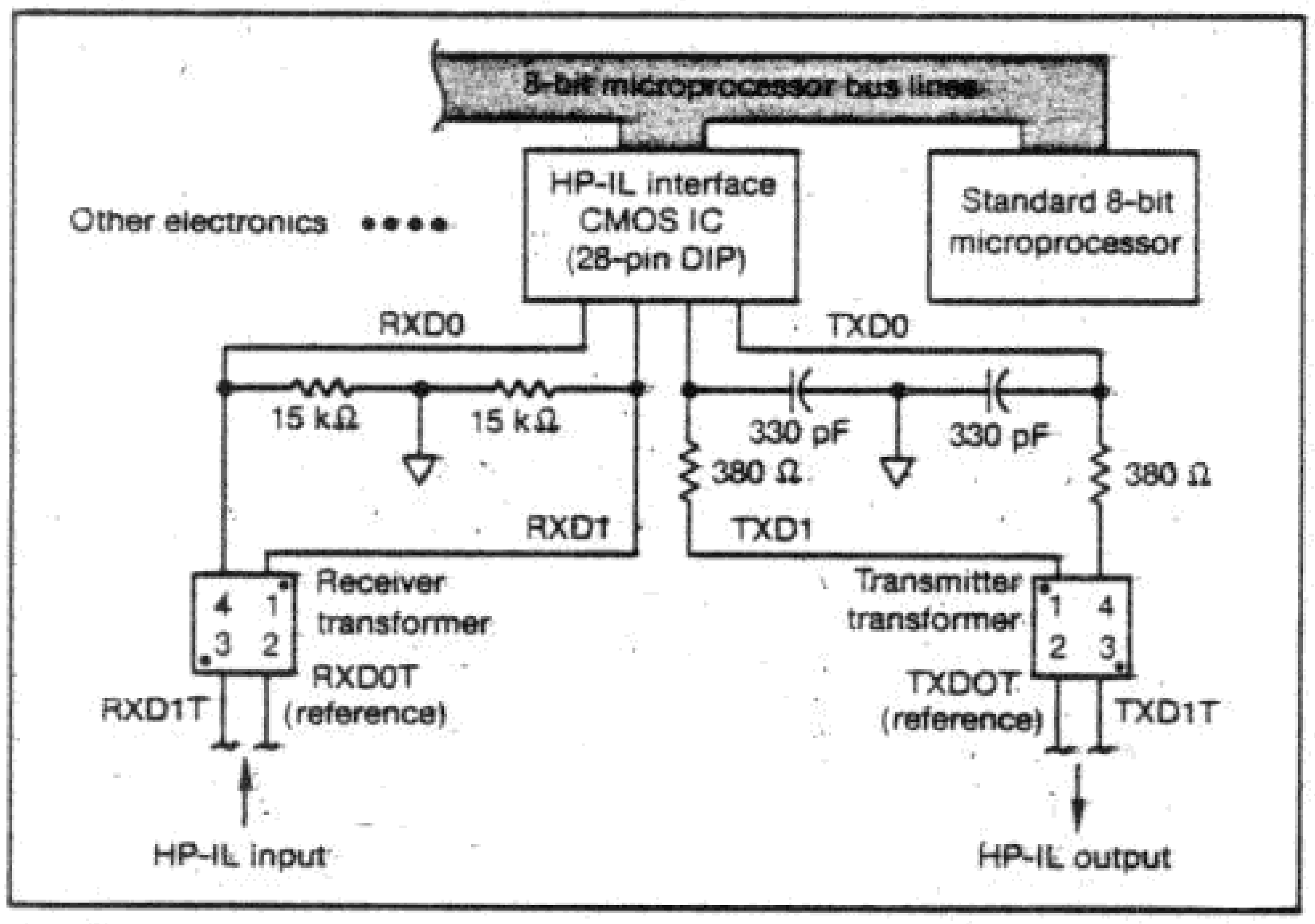
Through its control and status registers, the control logic stores operating information, implements various selectable operating modes, and controls the flow and interpretation of data within the converter.



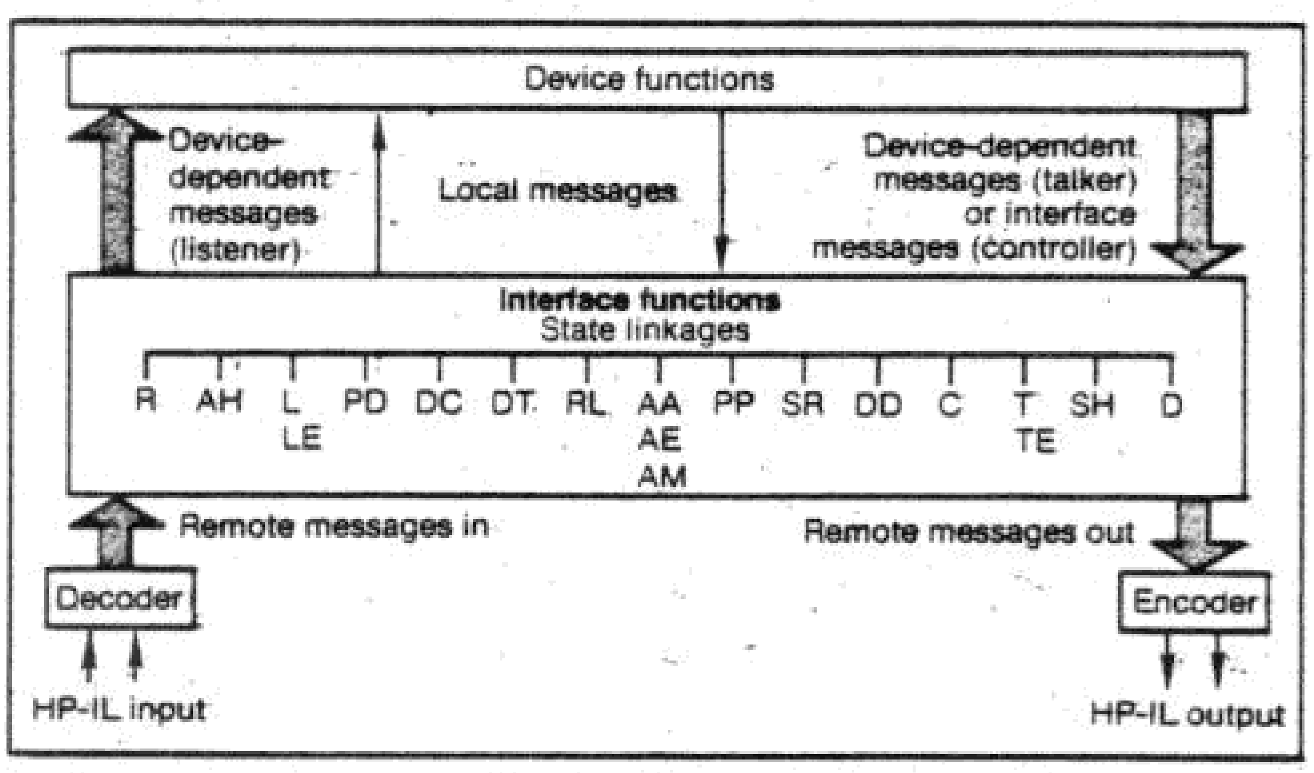
set as they pass on the frame.

Setting C_3 indicates an interface message. There are three groups of interface messages: command frames, ready frames (a special group that takes care of automatic addressing and other handshake functions), and identification frames (the controller's parallel-poll messages). Identification frames are used also to rapidly establish loop integrity. They carry the service-request bit, as do data frames.

The use of a CMOS IC for interfacing in HP-IL devices keeps the total interface power dissipation low. The IC also reduces system cost and size by eliminating multiple components and thus saving PC-board area, and by relieving the host computer system of having to keep track of some of the details of the protocols. In addition, a power-down function allows the loop controller to place all loop devices in a powered-down state and return them to operation when needed, further reducing overall power dissipation.



4. The use of transformer-isolated drivers and receivers is a key element in the HP-IL's isolation capabilities. The diagram above is an example of a simple circuit interfacing the loop with a device.



5. Communication on the HP-IL employs an 11-bit message frame. For simple addressing, the loop controller uses 5 bits to select talker or listener devices, with a maximum of 31 devices possible. If needed, devices can implement extended addressing for an address space of up to 961 devices.

The HP-IL data-transmission rate ranges from 3 to 5 kbytes/s, which corresponds to about one printed page per second and is well suited for many instrumentation and test systems. This speed is in keeping with present portable computers, which have internal memories of 2 to 8 kbytes. Even as the memory size of these computers grows to 32 to 64 kbytes, the HP-IL transmission rate will prove satisfactory.

In the future, the HP-IL will be capable of higher data rates. At present, it can drive 10 meters of zip-cord wiring between each device and 100 meters of higher-performance cable between devices.

Interfacing on the HP-IL

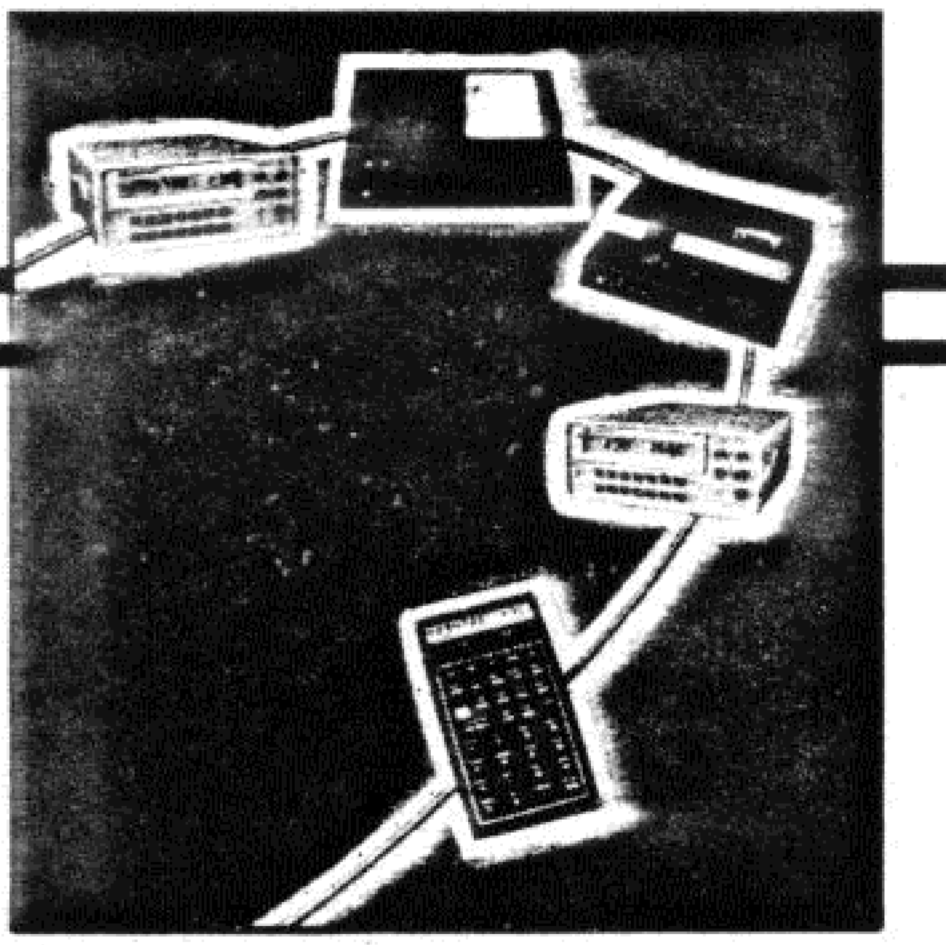
Inside an HP-IL product are two classes of functions: device and interface. The former are specific to each device and may be implemented by a designer in any manner. A voltmeter, for example, will have very different device functions from a calculator, or perhaps even from another voltmeter.

Interface functions, on the other hand, are those that permit a device to communicate via the HP-IL. The user chooses an appropriate subset of interface functions for a particular device, but they must behave in exactly the same, carefully defined manner as the same functions in all other HP-IL devices. This functional compatability is of extreme importance to guarantee proper communication and system integrity.

Addressing can be extended

The loop controller enables a talker or a listener by sending special commands that contain the address of the particular device. For simple addressing, 5 bits select the talker or listener. This length permits a maximum of 31 devices, since one of the codes is reserved to deselect all devices. If needed, devices can implement extended (two-frame) addressing, for up to 961 devices. With some care, simple- and extended-address devices can even be mixed on the same loop (Fig. 5).

When a listener receives a dedicated message



frame (i.e., one that does not affect other listeners), it passes it on with as little delay as possible, as noted. Whether or not the frame is immediately retransmitted depends on whether the device is actively operating as a loop controller, talker, or listener. The receiver (R), driver (D), and acceptor-and source-handshake (AH and SH, respectively) functions handle the proper reception, decoding, and transmission of messages.

A service-request (SR) function allows a device to signal the controller that it needs attention, even though it may not be addressed. As all devices use the same bit, the controller must poll the loop when it receives a service request, in order to find out which device needs service. If a faster response is required, the controller can execute a parallel poll (PP). This special message frame has 8 service-request bits. A device or group of devices is assigned by the controller to respond on one of these bits when it receives a parallel-poll message if it needs service.

The controller can switch a device's functions from front-panel to interface control, and vice versa, via a remote/local (RL) function. Through its device-clear (DC) and device-trigger (DT) functions, it sends commands that cause a device to enter a predefined functional state and begin an operation, respectively.

Automatic addressing

Because the HP-IL is a loop, the controller can automatically assign sequential addresses, either simple or extended, to all the devices. It does so by means of three functions —automatic address (AA), automatic extended address (AE), and automatic multiple address (AM).

Two extensions of the talker function should be mentioned. The device-identification function provides a device with the ability to respond to the controller with a string of data bytes containing the device's model number. The accessory-identification function is similar but returns only a single byte that indicates the generic device type: printer, controller, or mass storage, for example.

The HP-IL is virtually immune to errors, since the received frame is compared directly with the stored copy of the transmitted frame—an inherently reliable procedure.

Another noteworthy feature of HP-IL is that a block of controller command codes is reserved as device-dependent (DD) commands. The designer is free to use this block to control device functions (not interface functions) in any manner needed. Only

Interfacing the HP-41 computer with the HP-IL

The HP82160A HP-IL module links the HP-41 computer with the HP-IL system. It plugs into any one of the computer's four ports and consists of two 4-kbyte ROMs, a special HP-IL interface chip, several discrete components, isolation transformers, and HP-IL cables. The two ROMs direct the actions of the computer's microprocessor for functions determined by the user and for low-level manipulations of information being transferred through the loop.

Twenty-four functions are available to support mass-memory operations. The mass-memory system is based on a file-by-name directory system. Programs, data, key assignments, machine status, or the entire computer memory can be stored and retrieved by executing selected functions.

The user sends and receives information to and from HP-IL devices through 15 general-purpose control functions. For example, the INA function reads information from an arbitrary loop device into the 41's alpha register, and the OUTA function writes information from the 41's alpha register to an arbitrary loop device. These two functions provide the primary means of loop communications with the HP3468A digital multimeter and the HP82166 HP-IL converter. Among the other capabilities the interface module supports is the loop's power-down mode—an important feature for minimizing power consumption.

A chip within the module interfaces the computer's unique bus structures with the signals received and sent by the isolation transformers. Under direction of the ROMs, the computer's microprocessor reads and writes information to and from registers on the interface chip, which responds by transferring information to or from the loop.

devices active as talkers or listeners may respond to these commands.

Flexible mechanical design

The HP-IL is mechanically linked by a system of double-polarity panel receptacles, plugs, and connecting cables. The male-female structure of the plugs and receptacles defines the direction of information flow on the unidirectional loop: always out of male plugs and into female plugs. Thus it is impossible to misconnect devices (Fig. 6).

Both male and female plugs, as well as connecting to the panel receptacle, may be used to form running splices in the loop's hookup.

Because the HP-IL is intended to be easily reconfigurable, plugs and receptacles are held by detents instead of screws or clips. The detents are sufficiently strong to keep the loop connected under normal handling, yet they will disconnect under abnormal circumstances (as might occur if a device falls off the edge of a desk).

Since the HP-IL's connector system is housed in plastic, is detented instead of fastened, and is designed with wide tolerances for ease of manufacture, some relative motion of the contacts is bound to occur when the loop system is in operation. Because of this potential motion, the contacts are of the flexible, twist-pin type. Such contacts remain reliable through many connections and disconnections.

Using the HP-IL, systems can be implemented that are extremely easy to use. For example, suppose a loop contains an HP-IL computer and an 82162A thermal printer, as well as several other devices. When the user initiates the PRX (print the number in the X register) function, the computer automatically assigns addresses to all devices on the loop, polls each one by means of the accessory-identification function to find a printer, executes the

A thermal printer-plotter for the HP-IL

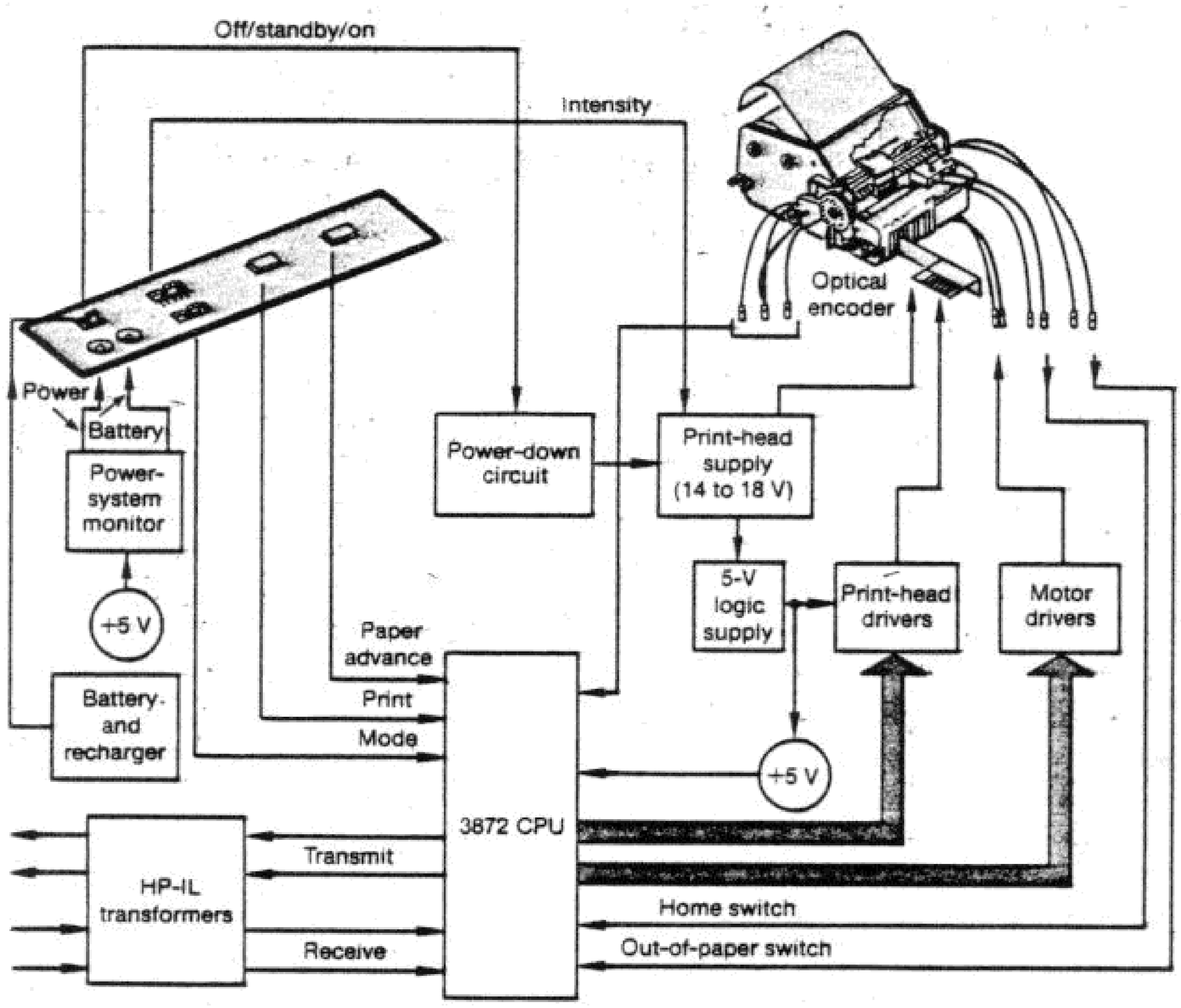
Designed expressly for the HP-IL system, the HP82162A printer employs a thermal print head and a dot-matrix format to print seven columns of 24 characters per line, 70 lines per minute. The printer is powered by a rechargeable nickel-cadmium battery (see figure).

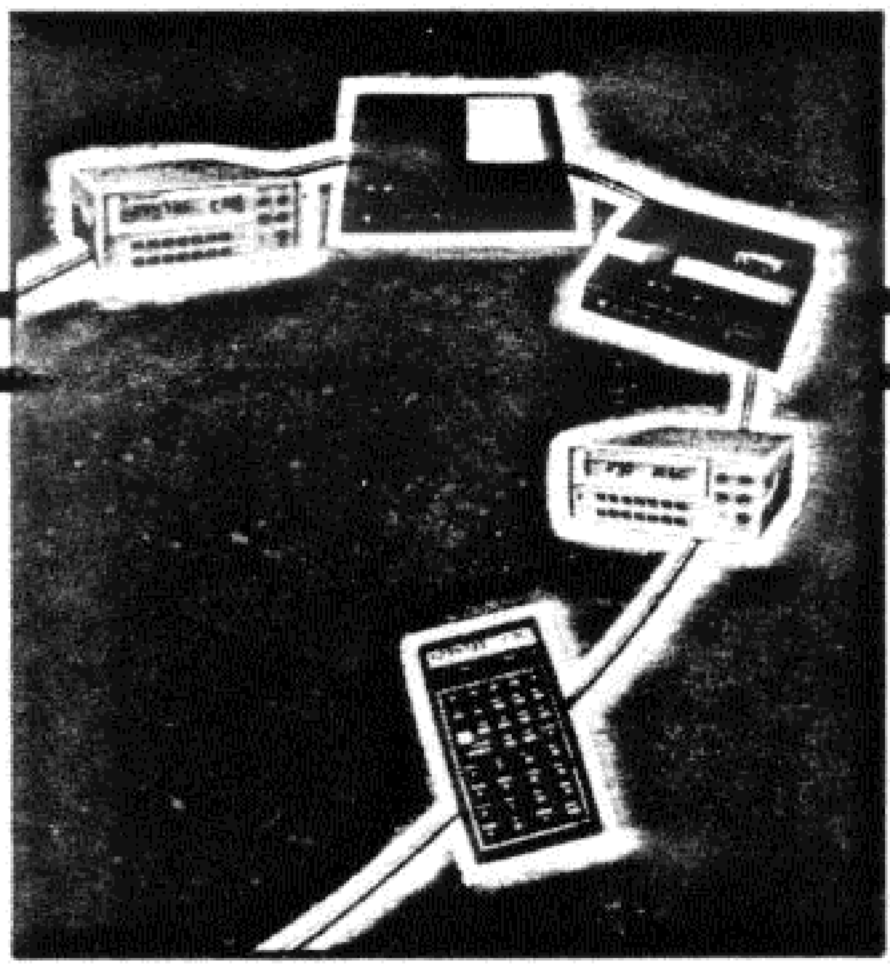
The printer is controlled by a type 3870 microprocessor with 4 kbytes of ROM and 128 bytes of RAM. Data over the interface loop is collected by the microprocessor's CPU and stored in a 101-byte buffer. When either a carrier-return signal is received by the buffer or the buffer overflows, the microprocessor turns on a dc motor. The motor, in turn, powers the printer's print head (which moves from right to left).

To ensure that each row of printed dots is properly aligned, positions are encoded by a scheme in which a reflective transducer generates a signal to interrupt the processor at equally spaced intervals across the paper. The microprocessor controls the motor's speed by turning it on and off to maintain a constant interval between transducer pulses.

The paper is advanced automatically on the return stroke of the print head. For consistent print quality, an internal timer controls the head's burn time. The printer also has a regulated 5-V power supply to power logic and a switch-adjustable 18-V supply for variable-intensity print settings.

Used with the right controllers, the machine is capable of graphics, double-width printing, parse-mode printing, and bar-code printing for the HP-41 computer. Also, it can power itself down upon command to prolong battery life.





proper command sequence to send the data from the X register to the printer, prints it, and then terminates the operation. The user need not worry about device addresses or the order in which the devices are connected. In fact, the user need not know anything at all about the HP-IL protocol: Devices can be added, removed, or reordered between PRX operations and the system will continue to function properly. In addition, software programs are easier to write, since they do not rely on specific system configurations. Such programs can utilize a mass-storage element and a printer, for example, without requiring specific devices.

An example of data transfer

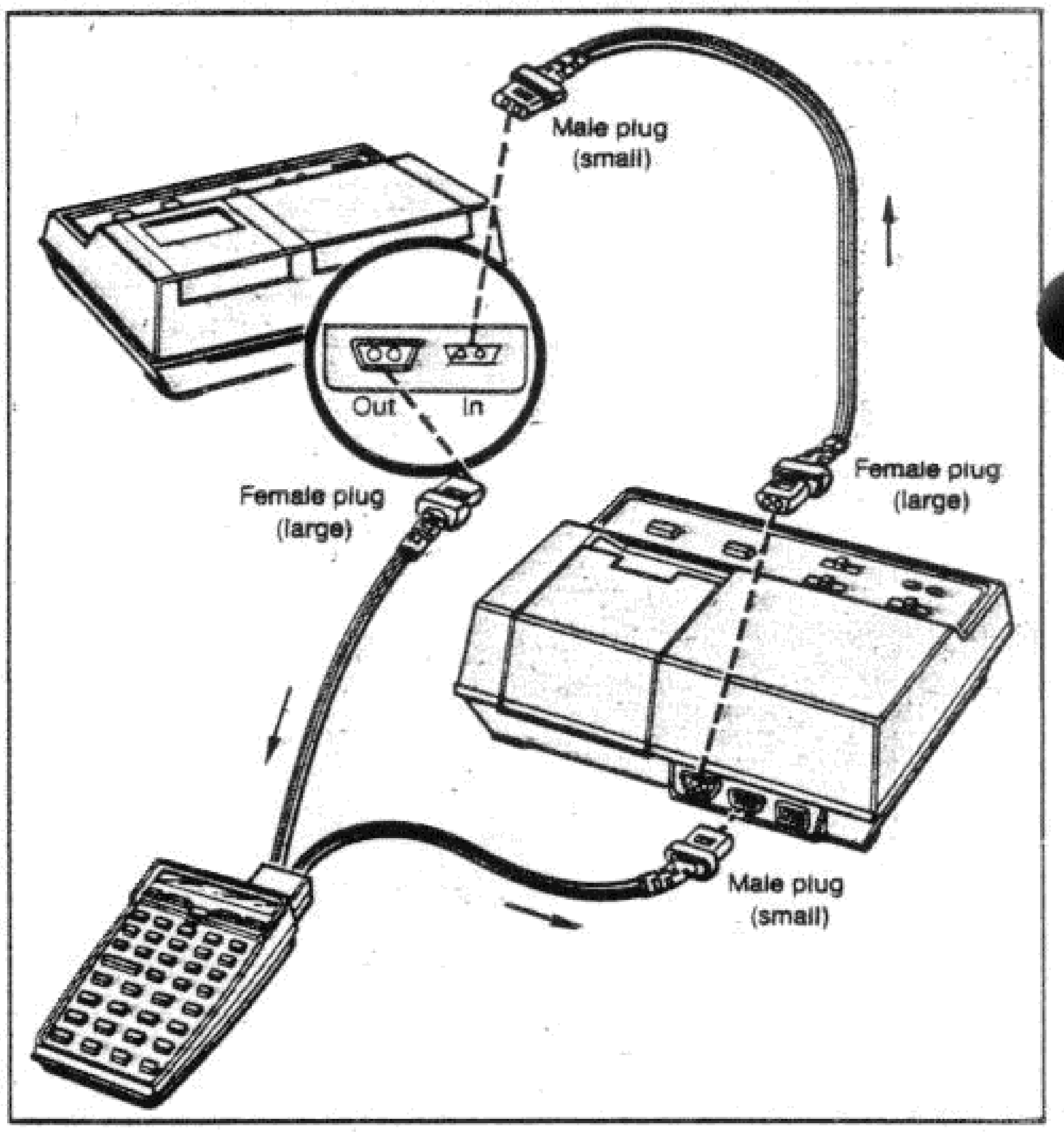
Suppose that a simple loop contains a computer, a printer, a cassette recorder, and a multimeter (Fig. 7). Assume that addresses have been assigned as shown, and that the multimeter has taken a reading that is held in its buffer. If the user executes functions from the computer to cause the reading to be printed, the following sequence shows what happens on the loop:

The computer initially sends the Unlisten command (UNL) to disable any previously active listeners. The command is immediately retransmitted by all loop devices and returned to the controller as the devices begin to execute it. The returned command is checked for errors, after which the controller sends out the RFC handshake frame. This message may be delayed briefly at one or more devices if they are not ready for another frame. The Listen Address 1 command (LAD 1) is next sent out on the loop, whereupon the printer becomes the active listener, and then the Talk Address 3 command (TAD 3), which causes the multimeter to be the talker. Each of these commands is immediately followed by an RFC frame to complete the handshake.

The controller now causes the transmission to begin by sending a special ready group frame called "Send Data" (SDA). The multimeter removes this frame from the loop and starts sending its data bytes. Its buffer might contain, say, the character string "+3.78VDC(CR) (LF)," each byte of which may be held up for an instant at the printer while it is loaded into the printer's buffer. It is then returned to the multimeter to complete the handshake and to be checked for errors. After that, the line-feed data byte causes the string to be printed.

After the multimeter checks the last byte, it sends an end-of-transmission (EOT) ready frame to the controller. The controller removes this message from the loop and is then ready for another command sequence.

To help a user understand the wide spectrum of capabilities and potential uses of the HP-IL, consider three possible applications. In the first, a university graduate student in forestry is working on his dissertation, a project studying the effect of various types of herbicides on underbush or tree growth. He makes observations and, on a 41 programmable computer, collects data from several different plots located miles away from school. He has written a short program for the machine that prompts him for

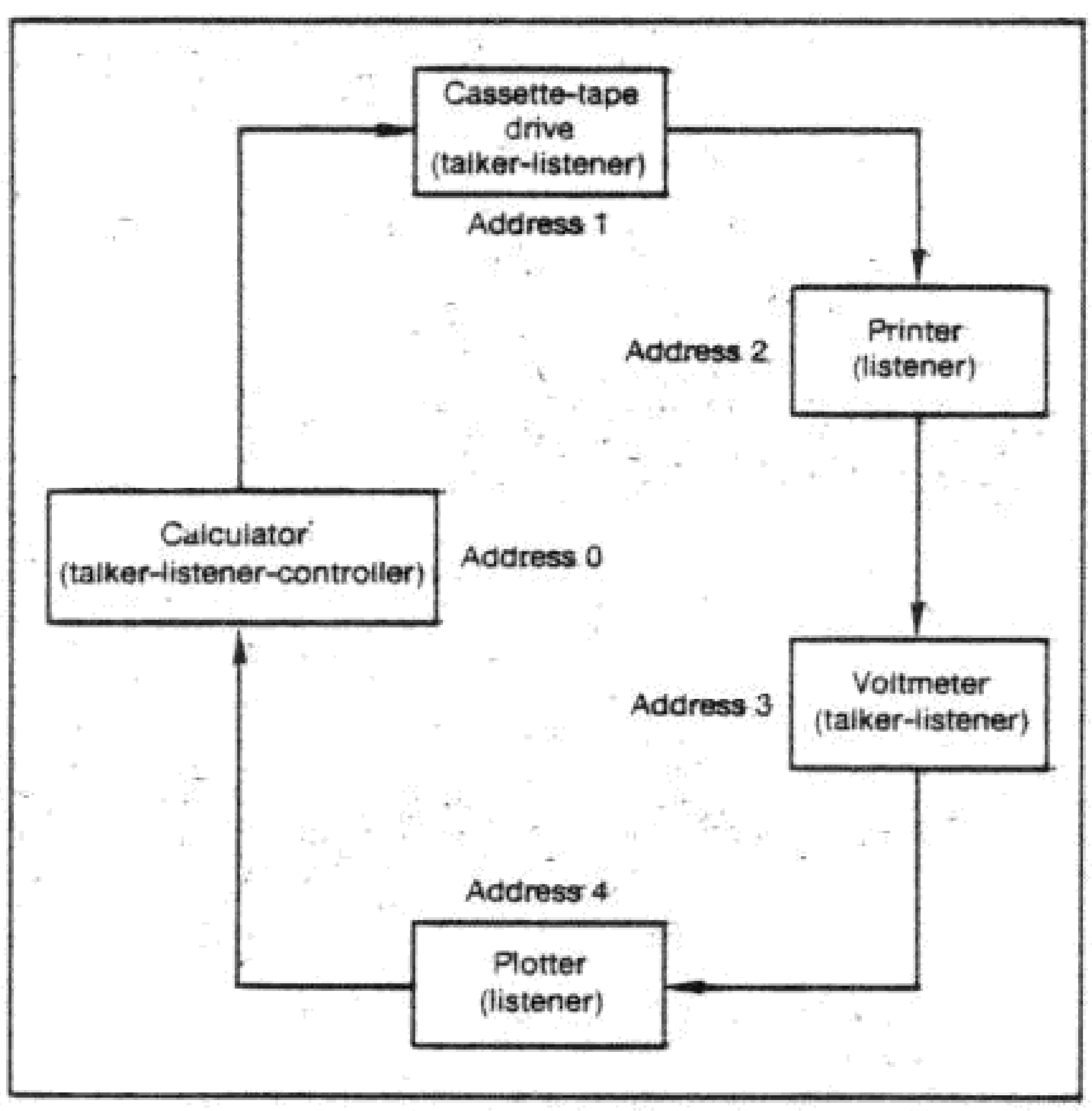


6. Since HP-IL male and female plugs are configured in a manner that defines loop information flow (information flows out of male plugs and into female plugs), devices cannot be hooked up improperly.

the correct data item and stores it, along with all the other data from each area, in separate internal registers.

The university's forestry laboratory has a small HP-IL system that includes the 82160A HP-IL module, which plugs into the HP-41; an 82161A digital cassette drive; and an 82162A 24-column thermal printer/plotter.

When the student returns to the laboratory, he plugs his computer into the system and inserts into the tape drive a minicassette containing his computer programs and the data he has collected on his project over a period of several months. Next he loads and runs a program from the minicassette that reads from the computer the data taken that day and stores it on tape with the rest of the data. He then loads another program that uses all the collected data on the tape to make finished graphic plots and histograms of various combinations of the data, like



7. A loop such as this one containing a controller, printer, tape drive, plotter and voltmeter is simple with the HP-IL. The calculator (controller) assigns each loop device an address as shown.

toxicity levels over time and the amount of undesirable foliage with respect to the type of herbicide.

Previously, the cost of such setup would have been prohibitive. Furthermore, the HP-IL's low power requirements and loop power-down feature permits unattended operation for extended periods. Had the student needed to measure toxicity level every hour for a week, for example, he could have left a small system in a weatherproof box at the sites, under battery power. The measuring apparatus would then have to have been connected to the HP-IL system, but that can be with an 82166A HP-IL converter.

It's good for small business, too

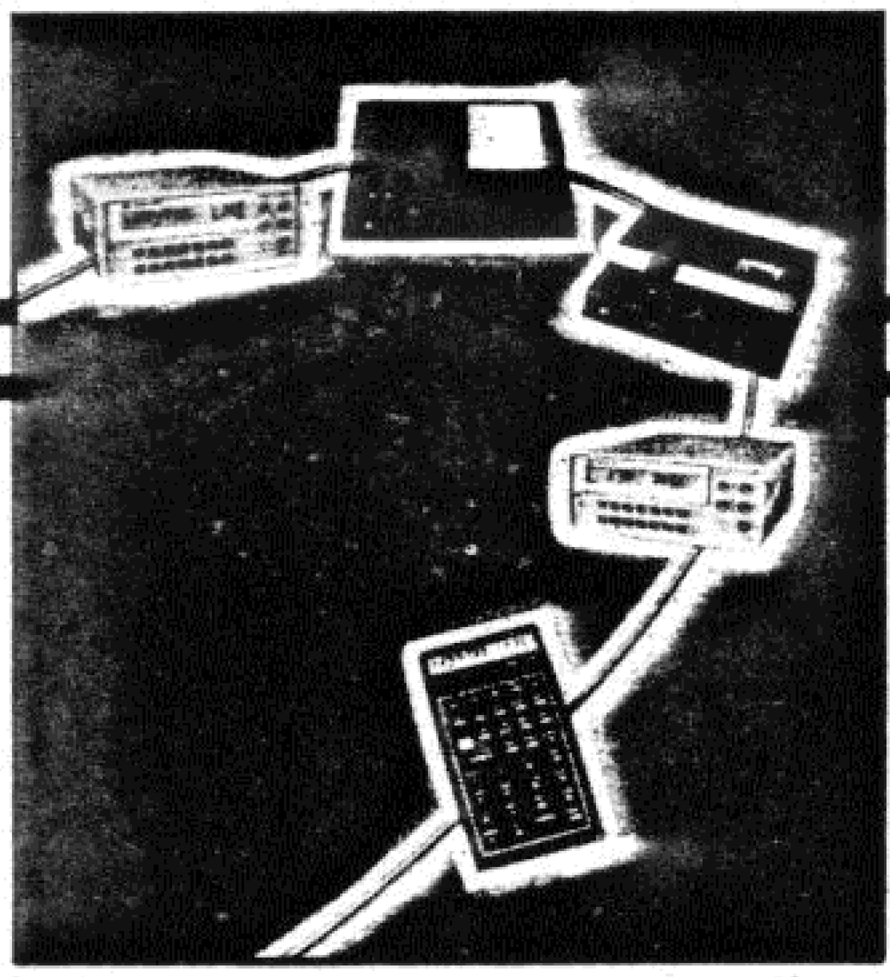
The second example concerns a small chain of three jewelry stores in nearby cities. Each store uses from two to five point-of-sale terminals consisting of an HP-41 computer with a custom keyboard overlay and an 82162A printer. These devices are connected via the HP-IL to two 82161A digital cassette drives and an HP 82905B 80-column printer with HP-IL option located in the back room of each store. Each computer contains a program that prompts the salesclerk for the necessary information for each transaction.

The data, consisting of item description, inventory number, price, charge-card number, and the like, is stored on one of the minicassette drives as each computer takes its turn controlling the HP-IL system. The small printer, located on the counter with the computer, prints the customer's receipt; simultaneously, the larger printer, in the back room prints a transaction summary.

At the end of the day's business, the manager places a master inventory tape in the second tape drive and runs a special program to update it based on the day's transactions in the first tape drive. From time to time, other special programs generate sales reports, a list of low inventory items to be reordered, and the like. Using the 82164A HP-IL-to-RS-232-C converter and a modem, the transaction or inventory data are transmitted over the telephone to the main store. Once again, until now such a system would have been much too expensive for most very small businesses.

Affordable automation

The third example is that of a small electronics firm that is about to begin volume production of a new audio-amplifier printed-circuit board. The production engineer has designed an automatic test system consisting of the HP-85A personal computer with the HP82901A disk drive, the HP-IL interface



module, and the HP3468A digital multimeter. The engineer has also built a special device containing a test fixture for the PC boards, a programmable power supply, a programmable waveform generator, and relay switches to connect these devices to the various PC-board test points. It also contains an 82166A HP-IL converter so that it too is controlled via HP-IL.

After initial debugging, the engineer writes the test program so that up to three identical test stations can work on the loop, since he has found that HP-IL and the 85A seem to be fast enough to support this throughput. Test programs and results are stored on the minifloppy disk drive for future failure analysis.

Later, when a second version of the amplifier board goes into production, simple program modifications will permit testing of both versions at the same time on different test stations on the loop. The low cost and flexibility of HP-IL are the deciding factors in the use of automated, instead of traditional manual, testing.□

A digital-cassette drive for the HP-IL

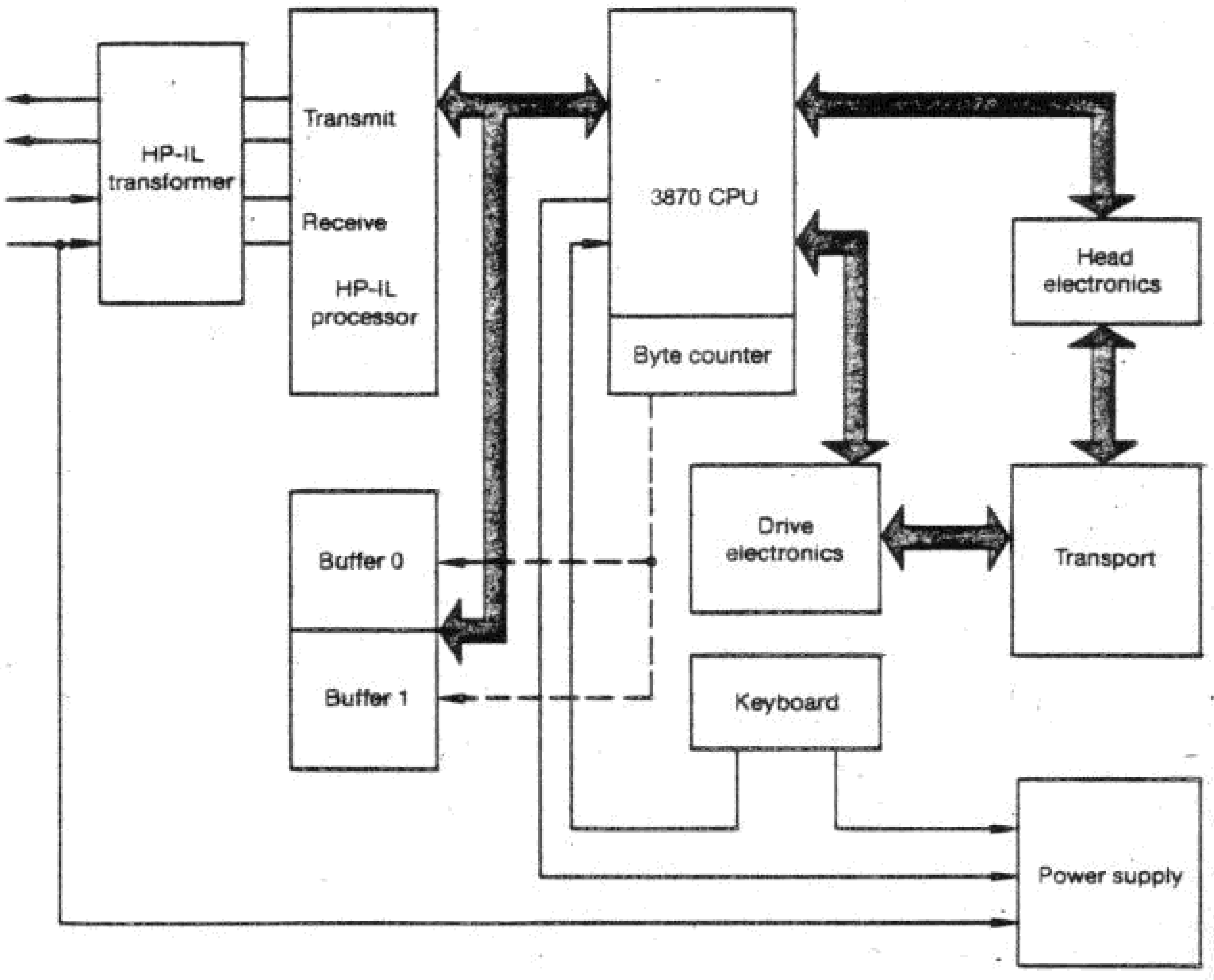
The HP82161A is a digital cassette drive with a mass-memory capability that makes data and programs accessible to the HP-IL system. Tape on the portable, battery-powered unit is moved by two hubs at 9 in./s for both reading and writing, and at 30 in./s for searching. A block diagram of the drive is shown below.

Reading and writing speed is controlled by averaging the back electromotive force of the drive's two hub motors, with speed variations being held to within ±3%. This tolerance allows the drive to store 131,072 bits of information on only a single 80-ft cassette tape, using biphase level-coding recording at 850 bits/in. The tape is formatted with two tracks of 256 records, each representing 256 bytes.

The drive is controlled through the HP-IL interface by device-dependent commands. The commands enable the loop controller to access any record on the tape randomly or to modify an individual byte in a record.

The tape directory is generated and maintained by the loop controller. A two-record buffer decreases tape motion in directory searches. One record is used for transfering information from the interface to the tape; the other, which acts as virtual memory for the controller, holds current the directory page.

The drive has a standby mode in which the loop controller can place it in a low-power, standby mode when it is not needed. It also features a storage compartment for two inactive tape cassettes.



DMM is first to work on automated interface loop

Designed for the HP-IL, this low-cost instrument offers automatic calibration, no-trimmer operation, and high resolution. A companion DMM is designed for the HP-IB.

The first major instrument to work on the Hewlett-Packard Interface Loop brings a lot more than automation to the solving of measurement problems. First of all, the HP3468A can be calibrated automatically. Equally important, it does away with trimmers, and it resolves 1 μV dc. But perhaps most important, it does all that for the startingly low price of $695.

However, the 3468A has not come alone into the world. Its companion, the HP3478A, boasts much the same virtues—automatic calibration, no trimmers, 5½ digits (300,000 counts). But it is designed to work in a different neighborhood—the HP-IB—and its $1300 price tag reflects that fact, as well as its greater sensitivity and higher speed.

Both instruments measure the five basic parameters: ac and dc voltage and current, plus resistance. They employ an integrating multislope analog-to-digital conversion technique that allows a user to trade off DMM reading speed for resolution (see ELECTRONIC DESIGN, June 7, 1980, p. 149). For example, the 3478A can take up to 71 3½-digit (±3000-count) dc-voltage readings per second (32 for the 3468A), and 3.7 5½-digit readings per second.

The Multi-Slope II conversion technique also makes possible high noise rejection (80-dB normal-mode noise rejection at 50/60 Hz in the 5½-digit mode). Both instruments employ a true-rms ac-voltage converter having a bandwidth of 20 Hz to 300 kHz and capable of handling signals with 4:1 crest factors full scale. An ohms converter is also included for two- and four-wire resistance measurement.

Each of the DMMs employs a 10-mm-high 14-segment 12-character alphanumeric liquid-crystal display with 12 annunciators. Calibrated electronically, the two instruments need no internal adjustments. All calibration can be performed either manually from the front panel (without removing the cover) or automatically from the back with an automatic-calibration system. Both feature sealed front-panel key switches, thanks to a single piece of silicone rubber.

Kenneth Jessen, Materials Engineer
Hewlett-Packard Co., Loveland Instrument Division
815 Fourteenth St. S.W., Loveland, CO 80537

A nonvolatile CMOS RAM

An internal CMOS RAM powered by a lithium battery for nonvolatility is the basis for calibrating the 3468A and 3478A electronically. In the 3468A, a 256 × 4-bit RAM stores a dozen gain and offset constants and can hold stored calibration constants for up to 10 years without any external ac power to the DMMs. The 3478A's RAM is even larger.

The user enables the calibration RAM by setting a rear switch, accessible through a slot in the meter's rear panel. This switch forms a fail-safe mechanism that prevents the user from accidentally changing the RAM's contents. The choice of a rear slot for the access allows, for example, metrology laboratories to seal the slot with a sticker, ensuring tamper-proof calibration.

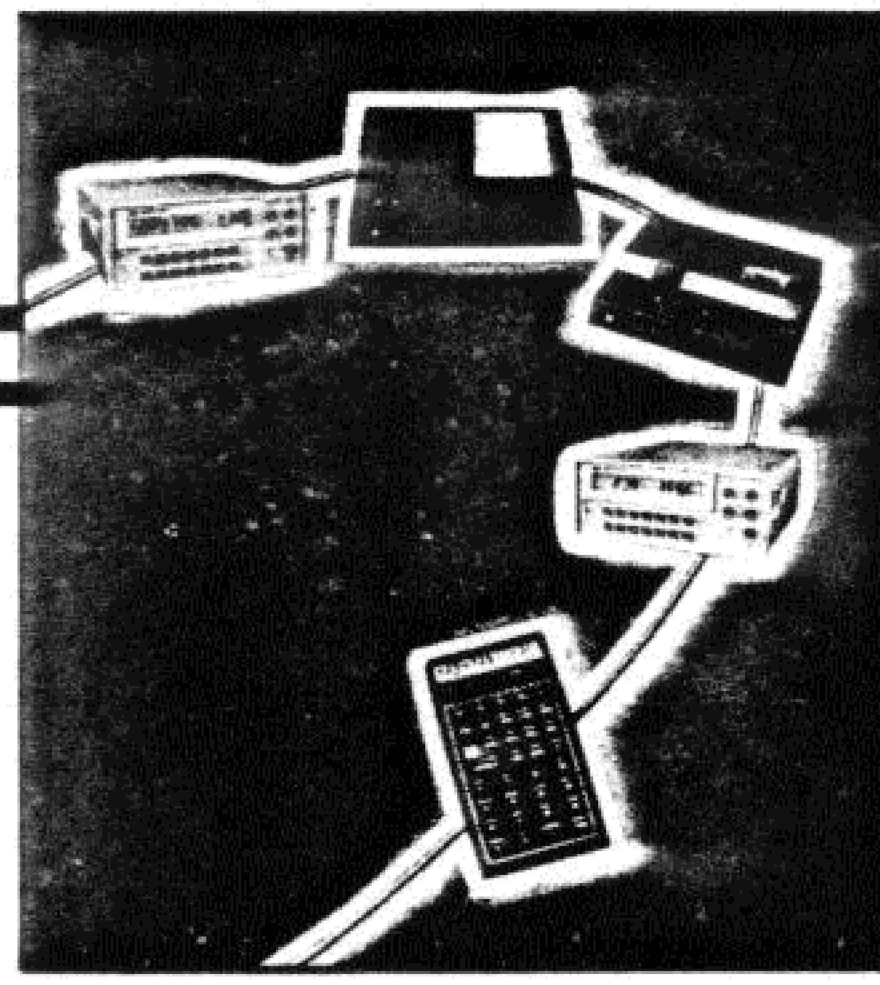
Once the rear switch is activated, the DMM displays a "C" on its front-panel LCD, alerting the user that it is ready to be calibrated. The user then selects a given function and range for calibration (see Figs. 1 through 3).

A known ac/dc calibration voltage or current, or a resistance, is applied to the DMM's input terminals. The DMM takes an average of 10 readings, computes the gain error and stores it in the CMOS RAM. The reading of the DMM is corrected according to the stored calibration constants. Full-scale calibration sources are not necessary. The known calibration source can be up to one-third of full scale, and the DMM can detect the difference.

Calibration of ac volts and current requires only a single 3-V, 1-kHz signal for full ac calibration. Both DMMs measure the 3 V on two different settings, and the internal microprocessor uses the readings to solve a pair of simultaneous equations to produce the gain and offset constants.

Plugging into the HP-IL

Because of its HP-IL interface capability, the 3468A (Fig. 4) can be used in a simple benchtop system under the control of an inexpensive controller such as the HP-41 hand-held computer. Along with HP-IL-compatible accessories such as the HP82162A thermal printer/plotter and the HP82161A digital-



cassette drive, the meter can be readily used for a host of new applications at economical prices. These include data logging, transducer-based measurements, and thermocouple linearization.

Linked in an HP-IL system, the 3468A and the 41 can store data and compute arbitrarily complex mathematical functions. Consider one such example.

Suppose a user wants to make the 3468A read in decibels instead of volts. Using the formula READING = 20 log X, a program can be written for the calculator and labeled "V to DB" (Fig. 5).

The first command, AUTOIO, configures the loop by assigning addresses to each device in the order in which they are connected (i.e., the first device is given address 1, and so on). This simple approach used in the HP-IL eliminates the need to know in advance what address a given device has.

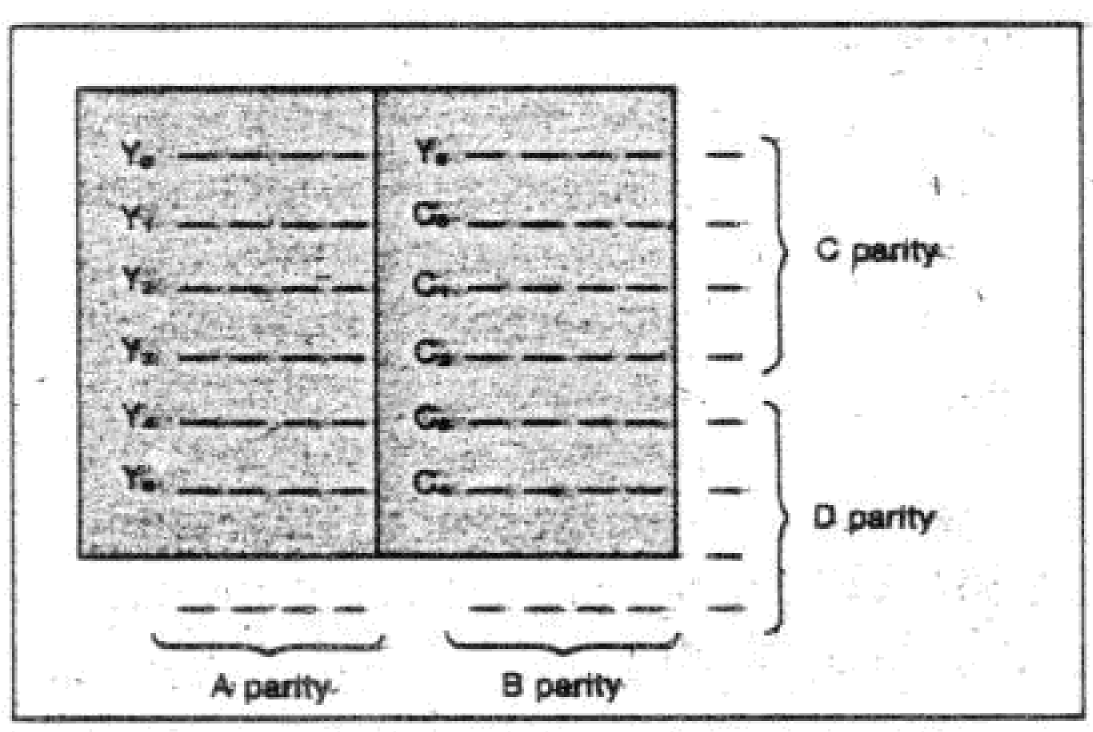
The next two commands, HP3468A and FINDID, are sent to each device on the loop. Each device is asked to identify itself. When the 3468A is queried, it transmits back to the 41 (which in this case is the loop controller) the statement HP3468A. This is compared with the HP3468A statement held in the computer's alpha register. The address of the 3468A previously assigned during automatic addressing is placed in the computer's numeric register, allowing the SELECT command to pick that device.

Next in the program, REMOTE tells the meter to ignore front-panel inputs and to accept only commands from the HP-IL. Then F1 RA N5 Z1 T1 instructs it to pick dc volts (Function 1), auto range (Range Auto), five-digit resolution (Number of digits 5), auto zero (Zero *1*) and continuous triggering (Trigger mode *1*). The OUTA sends these instructions to the 3468A from the 41's alpha register.

The next set of commands, IND, LOG, 20, *, tells the DMM to enter a decimal reading into the computer's X numeric register, take the log of it, and multiply the result by 20. FIX 1 is a format statement to produce a number that has one digit to the right of the decimal point. The command D2 (Display mode 2) performs a number of operations including blanking the normal voltage reading on the 3468A's display, then instructing the meter to accept the next 12 characters transmitted over the HP-IL and show them on its liquid-crystal display.

After the D2 command is sent out (with the OUTA command) and the computer's alpha register is cleared (with a CLA command), the ARCLX command is used to pull the contents of the X register (the result of the calculation 20 log X) into the alpha register. The next program line, " ⊣, space, space, space, D, B," merges the 20-log-X calculation with its units (dB). The OUTA command then places this line in the computer's alpha register and on the 3468A's display. Note that both the computer and the meter have compatible alphanumeric LCDs that are 12 characters long.

The 3468A DMM now behaves like a dB meter in-



1. So that it can be electronically calibrated, the HP-3468A 5½-digit multimeter—which is designed to work on the HP-IL—stores a dozen sets of constants (one set is shown) in an internal CMOS RAM. Each set contains a seven-digit offset and a five-digit gain correction. The last 2 bits of the D parity byte form the vertical and horizontal "parities of parity."

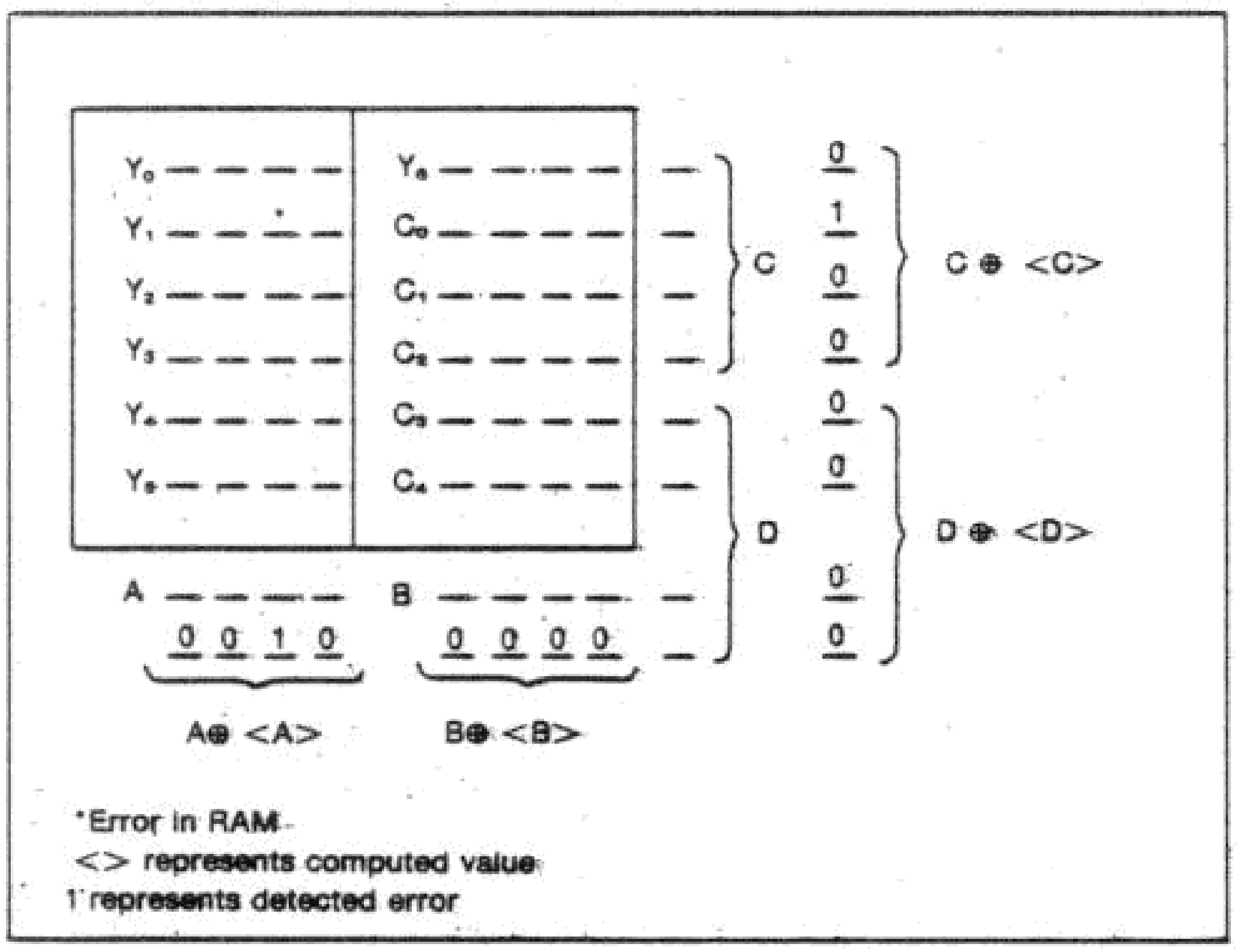
HP-IL, HP-IB—What's the difference?

Although HP-IL and HP-IB serve the same basic function—interfacing controllers, instruments and peripherals—they differ in many respects:

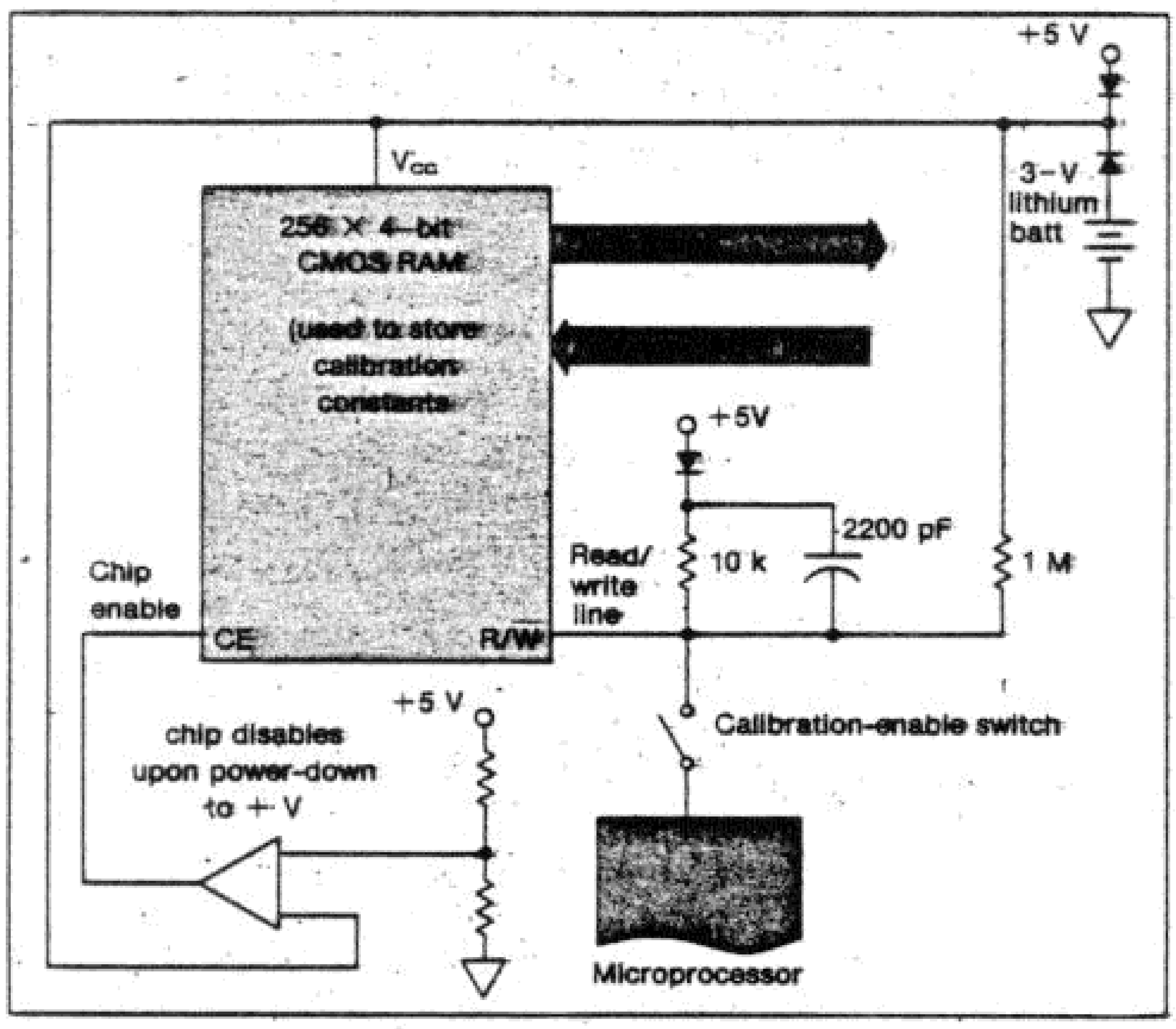
- HP-IL's lower power consumption lends it to portable, battery-powered systems. HP-IB generally does not apply in such cases.
- HP-IL system components generally will be low in cost and moderate in performance. HP-IB system components generally cost more and are located at the medium to high end of the performance spectrum.
- HP-IL systems work at relatively low data rates. HP-IB systems work at relatively high data rates.
- HP-IL allows device separations of up to 100 meters with shielded, twisted pairs (10 meters with zip cord). HP-IB requires extender hardware for long-distance connections.

stead of a multimeter. The only readings displayed are those in dB using 20 log X. The program (in Fig. 5) can be written for continuous readings by inserting a LBS 01 command ahead of TRIGGER and putting a GO TO 1 statement before NED.

Suppose the user wants to put together a simple system, in which the DMM takes readings that are printed by the printer. The computer configures the loop, either by a specific AUTOIO command or by turning on the power, such that the first device (the printer) is given the address 1 and the second device



2. To detect an error in a bit in Y_1, the computed A parity is exclusively ORed with the stored A parity, as are the values for the C parity. An internal microprocessor then corrects the error.



3. Stored calibration constants are protected by a circuit that prevents induced voltages from enabling the write mode. The constants can be changed only when the calibration-enable switch is closed.

(the DMM) the address 2. Assuming that the 3468A has been selected, its reading is placed into the computer's X (numeric) register via an IND program instruction. The instruction causes a Send Data command to be sent on the bus to the meter. The DMM does not retransmit the command but begins instead to send its reading to the controlling computer as a sequence of frames ending with carriage-return and line-feed characters. As the computer receives the reading, it loads it into its X register. Thanks to the loop structure, the DMM can check for errors before transmitting the next data frame. Should an error be detected, the reading sequence is terminated by an End of Transmission with Error statement. When the computer receives this statement, the program is halted and the computer's display reads TRANSMIT ERR.

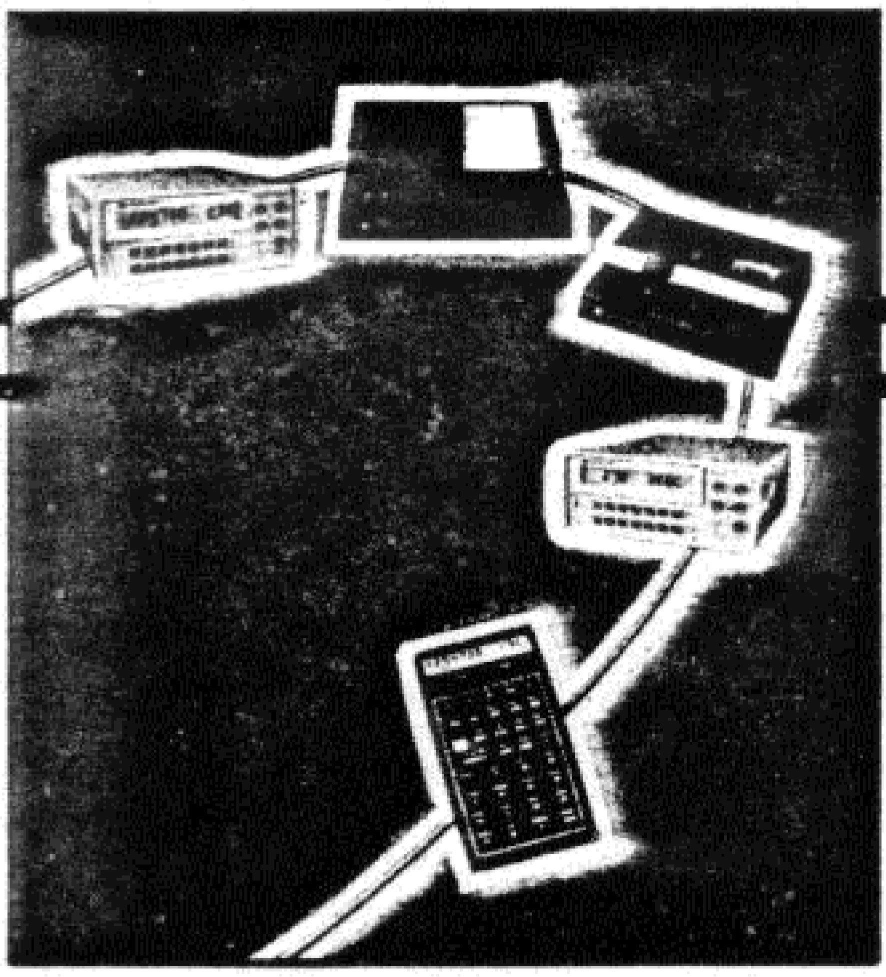
A PRX instruction sends the reading to the printer. The computer polls each loop device, beginning with the first one, until the printer is found. It then commands the printer to become a listener, whereupon the meter's reading, held in the computer's X register, is sent out as a sequence of frames. The printer accepts the data, storing it in its own buffer memory. At the end of transmission, carriage-return and line-feed characters are sent. Upon seeing these characters the printer prints the contents of its buffer memory as a single line.

The speed of this system is limited by the speed of the computer, which is about 2 readings per second. The meter can take up to 32 readings per second (3-½ digits), below the maximum transmission speed of the HP-IL.

Consider now the procedure of writing a number of useful HP-IL programs.

The following initialization program locates the 3468A DMM in the loop (finds its address), selects it to receive information, and puts it into the remote mode. The program should be run before any other command or program is executed; however, it need be executed only once unless a device is turned off or the loop is broken into to insert or remove a device. The HP82160A HP-IL module must be inserted in the calculator before entering the program:

Keystrokes	Display
	01 LBL INI68
INI68	02 AUTOIO
AUTOIO	03 CF17
17	04 HP3468A
HP3468A	05 FINDID
FINDID	06 SELECT
SELECT	07 REMOTE
REMOTE	08 RTN



The initialization program has now been entered. If the meter has not yet been connected to the loop, it should be done now. Remember, the computer must be turned off before connecting the DMM. The initialization program will not be lost in the computer with power turned off.

To run the initialization program, press INI68. The computer display will return with the HP-IL address of the 3468A. In addition, the REM (remote) annunciator will turn on to indicate that the meter is ready to accept commands from the calculator. Going one step further and assigning this program to the computer *SUM*+ key will make execution of the program easier in the future. Press IN168. Now, to execute the initialization program when necessary, put the computer in the user mode and press the ( ) A#+ keys.

One approach to program writing involves building-block subprograms that simplify the task. With this approach, a mainline program can be written that uses already existing subprograms. The following program contains seven measurement subprograms and an eighth subprogram that sets up the 3468A for the measurement, triggers it, and reads the measurement data to the computer. The measurement subprograms are used to specify the type of measure-

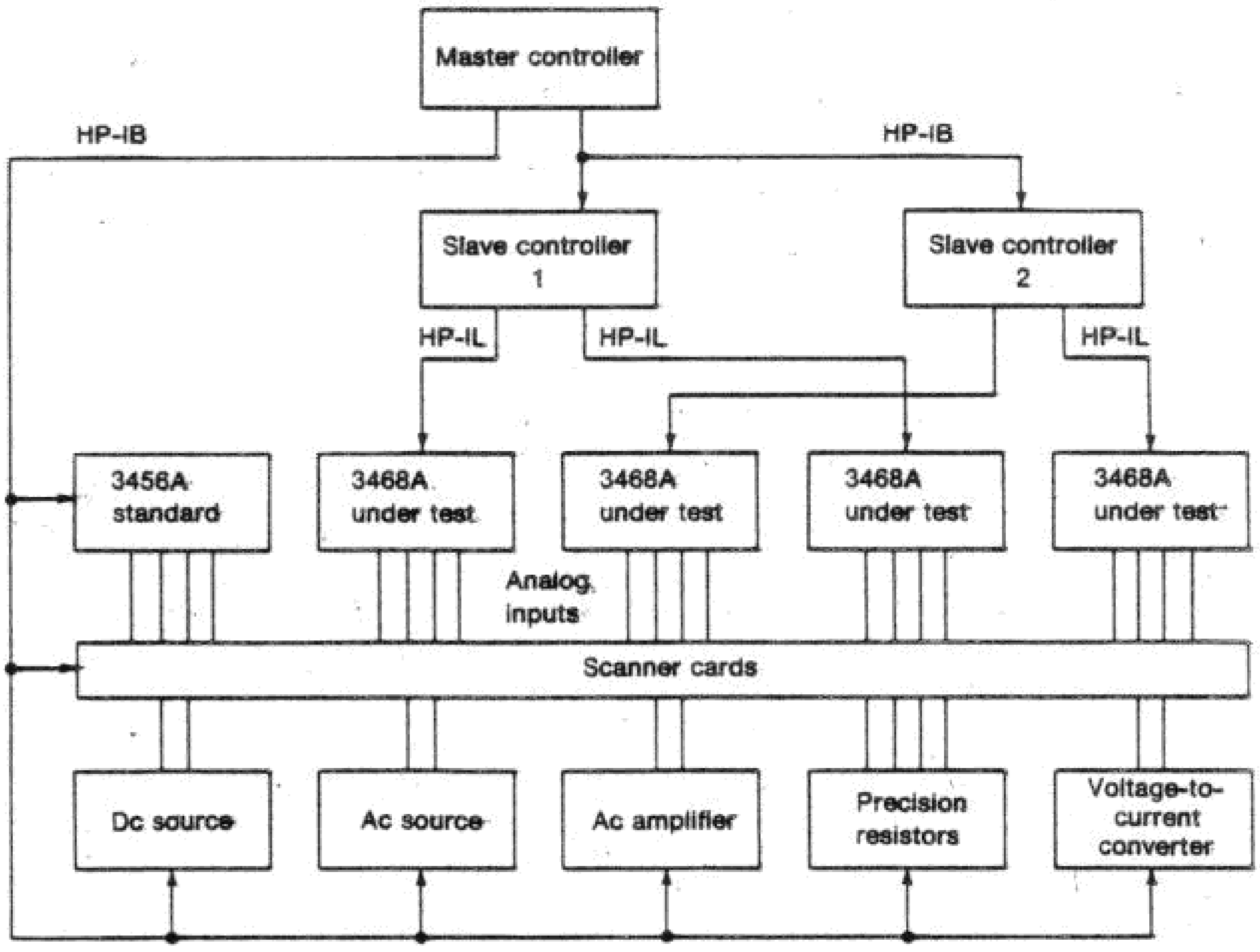
DMM is automatically calibrated on the production line

Electronic calibration has allowed Hewlett-Packard to automate its production line for the HP3468A 5½-digit multimeter to the extent that the only human intervention required is to check the instrument's display and to load and unload the robotic test fixture. The test fixture, which handles two meters at a time, makes all the required connections, including inserting the power cord and depressing all the front-panel keys. Desktop computers built into the test station handle all testing and calibration functions.

The test station (see figure) uses an HP3456A six-digit voltmeter as the calibration standard. The entire system is automatically verified for accuracy according to this single standard at the beginning of a production run. The voltmeter makes dc voltage measurements accurately to within 10 ppm and with sensitivity in the last digit down to 100 nV.

Other instruments in the test station include a precision dc voltage source, an ac voltage source, a bank of precision resistors, and a voltage-to-current converter. If small offset errors within acceptable limits are discovered while the test station is being verified, they are stored for later use during instrument tests to correct the sources. If, on the other hand, offset errors are found to be outside acceptable limits, the operator is notified of a system failure.

Each test station contains two fixtures so that one pair of instruments can be tested while the other fixture is being unloaded and reloaded with untested instruments. Putting a pair of 3468As in a test fixture reduces the effect of settling time imposed by the precision standard sources on the throughput of the station. As a source is settling while connected to one meter, the other instrument is being calibrated.



ment data to the computer and to tell the meter the type of measurement it is to make. They then branch to the eighth subprogram.

Do not erase the INI68 program before loading the building-block program. Loading is done by executing the following command string before any other program is entered:

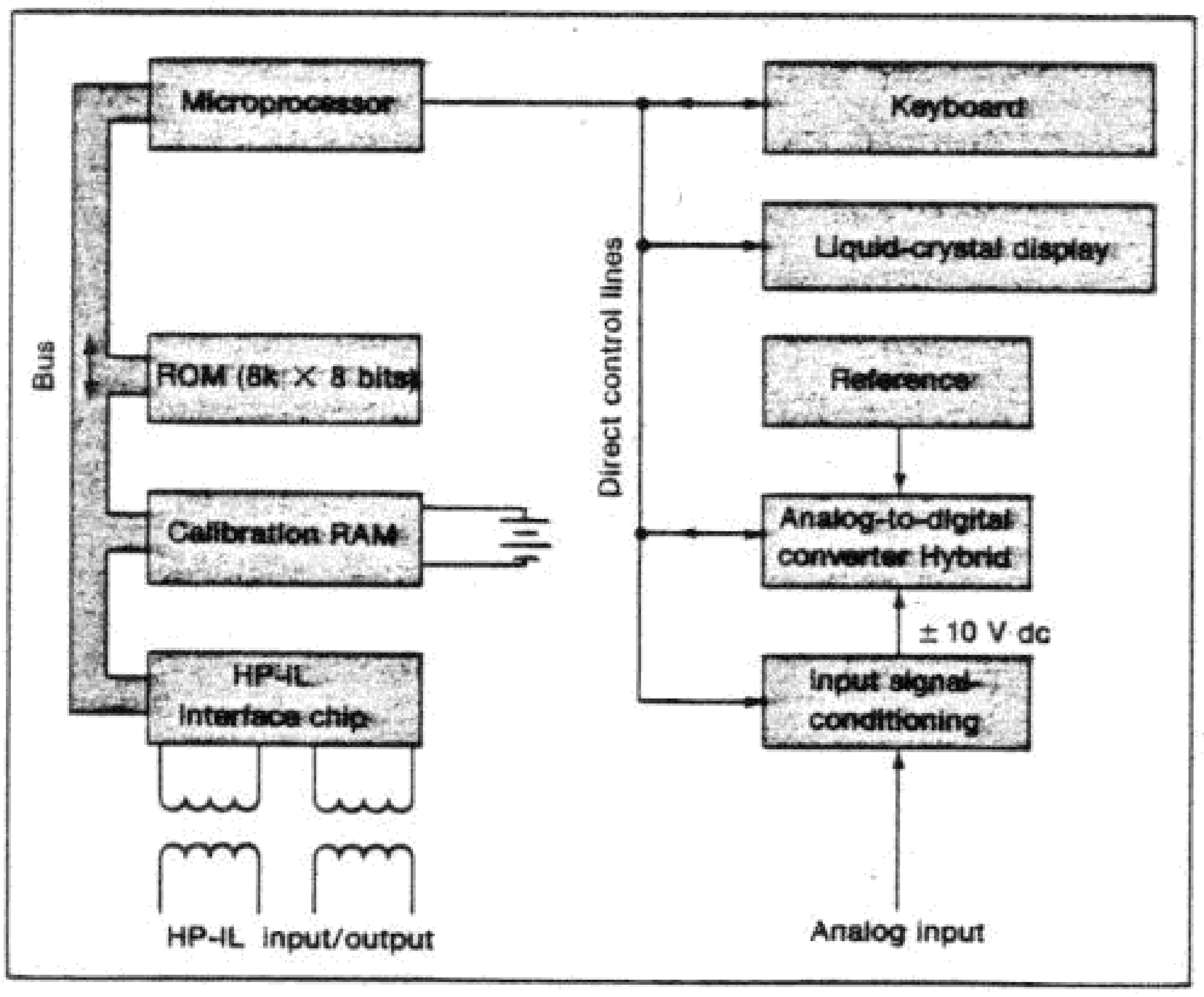
```
01 LBL "BLOCKS"
02 LBL "MVDC"
03 "F1"
04 GTO 00
05 LBL "MVAC"
06 "F2"
07 GTO 00
08 LBL "M2OHM"
09 "F3"
10 GTO 00
11 LBL "M4OHM"
12 "F4"
13 GTO 00
14 LBL "MADC"
15 "F5"
16 GTO 00
17 LBL "MAAC"
18 "F6"
19 GTO 00
20 LBL "MXOHM"
21 "F7"
22 LBL 00
23 " ⊣ RAZ1N4T2"
24 OUTA
25 IND
26 RTN
27 .END.
```

Temperature measurements

The TEMP program computes the temperature, in degrees Celsius, corresponding to the resistance of a thermistor. The program has been designed to work with thermistors exhibiting a 5,000-kΩ resistance at 25°C, such as the type 44007 or its equivalent. range of −80° to +150°C:

The program is written for two-wire resistance measurements and provides accurate results if the thermistor is used at a temperature at which its resistance is much greater than the resistance of the test leads. For greater accuracy, a four-wire resistance measurement should be made. To change from a two- to a four-wire mode, change line 15 to read XEQ "M4OHM." The program is useful over a range of −80° to +150°C:

```
01 LBL "TEMP"
02 -6.760961227
03 STO 01
04 5314.3107
05 STO 02
06 322.807684
07 STO 03
08 -5.952179428
09 STO 11
10 4751.384293
11 STO 12
12 303.33182
13 STO 13
14 XEQ "INI68"
15 XEQ "M2OHM"
16 LBL "MEAS"
17 3134
18 ENTER↑
19 TRIGGER
20 IND
21 STO 00
22 X>Y?
23 GTO 35
24 RCL 13
25 RCL 12
26 RCL 11
27 GTO 36
28 LBL 35
29 RCL 03
30 RCL 02
31 RCL 01
32 LBL 36
33 RCL 00
34 LN
35 X<>Y
36 -
37 /
38 X<>Y
39 -
40 STO 00
41 FIX 2
42 "D2"
43 RCL 00
44 ARCL X
45 "    DEG C"
46 OUTA
47 GTO "MEAS"
48 .END.
```

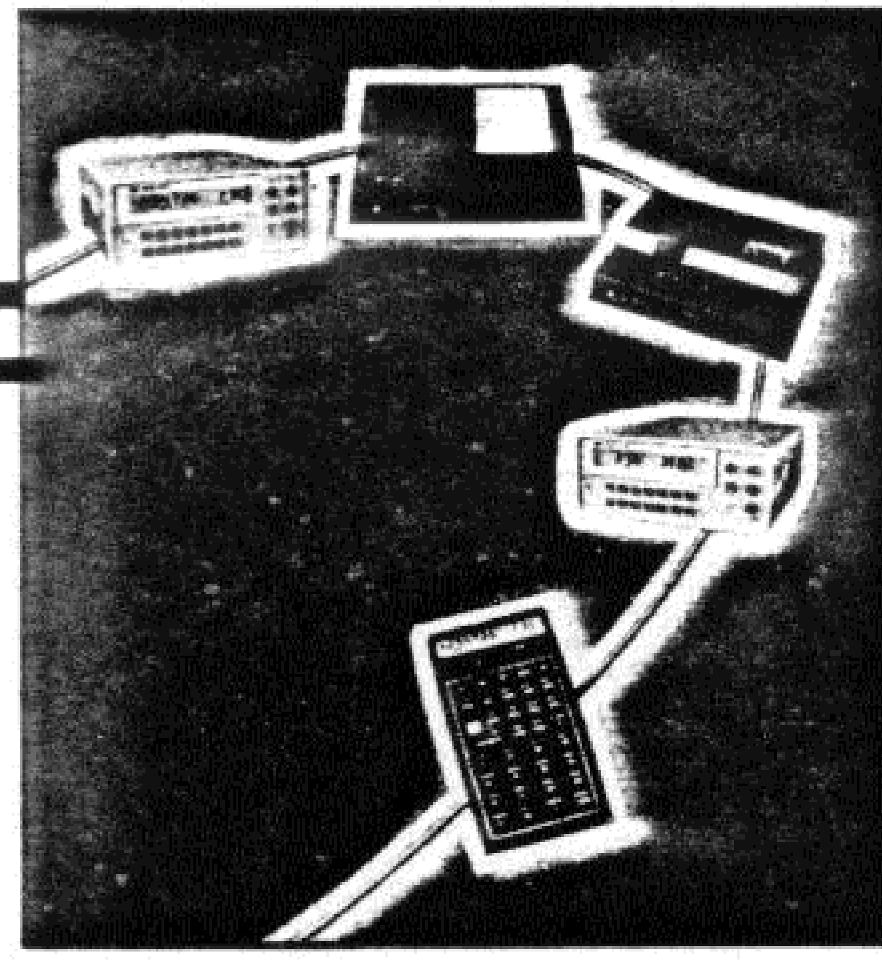


4. When the 3468A DMM is connected to the two-wire interface loop system, some means is needed to avoid separate floating- and earth-grounded logic. Isolation transformers do the job.

Data logger

The following data-logger program (LOGGER) illustrates the power and flexibility of an HP-IL system. The program takes ten sets of ten readings each, for a total of 100 readings, and stores them on a cassette for future reference. The program can easily be enhanced by adding a linearization routine between lines 15 and 16. An example of a linearization program is the previous program that converted the resistance of a thermistor into a temperature. In this way, temperature rather than resistance measurements are stored on the tape.

The READ program demonstrates how easily data can be read from the cassette tape and listed on a printer. Again, this program can be enhanced by adding a routine to plot the data on the printer. Of course, the data can also be printed or plotted in real time as the measurements are being taken in the LOGGER program.



For the LOGGER program to run, the interface loop connections must be made between the meter, computer, cassette-tape drive, and printer, all of which must be turned on.

Before data can be stored on the cassette tape, the tape must be initialized. That is done by executing the (NEWM) new medium command.

Line 03 executes the MVDC subprogram, which can be changed to suit particular needs. Deleting the line entirely allows the 3468A to be set manually as often as needed. When the device to be tested has been connected to the meter's input terminals, typing in LOGGER gets one ready for testing.

```
01 LBL "LOGGER"
02 XEQ "INI68"
03 "DATA"
04 100
05 CREATE
06 0
07 SEEKR
08 1.010
09 STO 00
10 LBL 01
11 11.020
12 STO 01
13 LBL 02
14 TRIGGER
15 IND
16 STO IND 01
17 ISG 01
18 GTO 02
19 11.020
20 WRTRX
21 ISG 00
22 GTO 01
23 .END.
```

```
01 LBL "READ"
02 "DATA"
03 0
04 SEEKR
05 1.010
06 STO 00
07 LBL 01
08 11.020
09 READRX
10 11.020
11 PRREGX
12 ISG 00
13 GTO 01
14 .END.
```

Space for storing the 100 readings must be allocated on the cassette tape. This space (or file) is created and given the name DATA in lines 03, 04, and 05. Lines 06 and 07 are used to return to the beginning of the DATA file on the cassette, for data storage.

Register 00 is used as a loop counter to take ten sets of readings. The value 1.010 is stored in register 00 (lines 07, 09) and it is incremented and tested at line 21.

Register 01 also acts as a loop counter (lines 11, 12), but counts ten readings per set (register 00). The value 11.020 stored in register 01 specifies registers 11 through 20 as temporary storage for the measurement data. Line 14 triggers the 3468A and line 15 enters the results. Line 16 stores the data in the register pointed to by register 01. Register 01 is incremented after each reading to point to the next empty register in line 17.

Register 01 is also tested in line 17. Remember that program line 11 specified register 20 as the last register to be used. If register 01 points to register 21, the program prepares to store the ten readings on the cassette. Line 19 calls out register 11 through 20 as data-storage registers. The WRTRX command (line 20) takes the data from the storage registers and then writes them onto the cassette tape.

Register 00 is tested in line 21 to see if ten sets of ten readings each have been made. If not, the program loops back (line 22) to label 01 (line 10). When the full ten sets of measurements have been completed, the program automatically comes on end.□

01	LBL "V TO DB"	(label for program)
02	AUTOIO	(automatic addressing; also done when the power is first turned on)
03	"HP3468A"	(finds the DVM in the loop by sequentially ask - ing each device for its ID)
04	FINDID	
05	SELECT	(DVM is selected as the device 41C will talk to)
06	REMOTE	(tells DVM to accept only HP-IL input)
07	"F1 RA N5 Z1 T2"	(picks DCV, auto range, 5 digit resolution, auto zero and continuous triggering; this information is put in the alpha register)
08	OUTA	(Output Alpha sends these instructions to DVM)
09	TRIGGER	(optional - but sends a trigger,; for continuous readings, the program can be made to loop to line 09)
10	IND	(Input Decimal puts the DVM reading in the X register)
11	LOG	(take log of X)
12	20	
13	*	(multiply by 20)
14	FIX 1	(the number is to have one place to the right of the decimal point)
15	"D2]	("D2" is put in the alpha register and when sent, will blank the DMM's display of its normal reading and take the next 12 characters and put them in the display instead)
16	OUTA	(Sends the "D2" command to the 3468A)
17	CLA	(clear the alpha register)
18	ARCL X	(Alpha Recall X pulls the contents of the X register into the alpha register)
19	" DB"	(entered as ", space, space, space, D, B" to combine the contents of the X register with DB)
20	OUTA	(output the above to the 3468A's display)
21	END	

5. As part of a team effort, the DMM reads voltages and sends the values over the HP-IL to the calculator, which converts the information into decibels. Note that the calculator-controller finds the DMM on the loop automatically by using the automatic addressing feature.