JANUARY 1975

# HEWLETT-PACKARD JOURNAL

# The Hewlett-Packard Interface Bus: Current Perspectives

*First announced over two years ago, the Hewlett-Packard Interface Bus has undergone refinements that make it suitable as a model for a proposed international standard.*

**by Donald C. Loughry**

THE GOAL OF AN INSTRUMENTATION system is to monitor some process or perform measurements on a device. An interface system, a means toward this goal, provides the essential communications link between the components of the system. No single interface method is a panacea for all of the world's interface requirements and the HP Interface Bus system is no exception, but it does fulfill major needs for a wide range of calculator and computer controlled instrumentation systems.

The HP Interface Bus is a definition of an inter-device connection scheme that is optimized for programmable bench instruments. It is applicable as well to other components essential in instrumentation systems.

An interface system definition has three primary elements: mechanical specifications (cables, connectors, etc.), electrical specifications (voltage and current levels for transfer of signals), and functional specifications (a precise definition of all the signal lines, the protocol and timing relationships for using the lines, and the repertoire of messages that may be exchanged).

A fourth interface system element involves operational characteristics and specifications. These tend to be device-dependent characteristics, such as specific programming codes unique to each instrument, and perhaps system-dependent characteristics such as the application software. The primary focus of an interface definition, however, is away from device-dependent characteristics and toward mechanical, electrical, and functional specifications that are device and system independent. This approach leads to a feasible and broadly useful interface system definition.

The HP Interface Bus is outlined briefly in the box on the opposite page. It is a byte-serial, bit-parallel, partyline bus structure organized to provide communication among a group of up to fifteen instruments and system components. Messages are exchanged over the bus asynchronously and flow both to and from a given instrument.

**Cover:** *As more devices are designed to work with the HP Interface Bus, it becomes easier to assemble low-cost systems, as HP product manager Jane Evans is doing here. Jane (BS Chem, BSEE, MBA) has been bringing ten years of HP instrumentation and data systems experience to bear on use of the HP Interface Bus in solving problems in automatic measurement systems.*

**In this Issue:**

Printed in U.S.A.

## The HP Interface Bus

The HP Interface Bus transfers data and commands between the components of an instrumentation system on 16 signal lines. The interface functions for each system component are performed within the component so only passive cabling is needed to connect the system. The cables connect all instruments, controllers, and other components of the system in parallel to the signal lines.

Eight of the lines (DIO1-DIO8) are reserved for the transfer of data and other messages in a byte-serial, bit-parallel manner. Data and message transfer is asynchronous, coordinated by the three handshake lines (DAV, NRFD, NDAC). The other five lines are for control of bus activity.

Devices connected to the bus may be talkers, listeners, or controllers. The controller dictates the role of each of the other devices by setting the ATN (attention) line true and sending talk or listen addresses on the data lines (D101-D108). Addresses are set into each device at the time of system configuration either by switches built into the device or by jumpers on a PC board. While the ATN line is true, all devices must listen to the data lines. When the ATN line is false, only devices that have been addressed will actively send or receive data. All others ignore the data lines.

Several listeners can be active simultaneously but only one talker can be active at a time. Whenever a talk address is put on the data lines (while ATN is true), all other talkers will be automatically unaddressed.

Information is transmitted on the data lines under sequential control of the three handshake lines. No step in the sequence can be initiated until the previous step is completed. Information transfer can proceed as fast as devices can respond, but no faster than allowed by the slowest device presently addressed as active. This permits several devices to receive the same message byte concurrently.

The ATN line is one of the five control lines. When ATN is true, addresses and universal commands are transmitted on only seven of the data lines using the ASCII code*. When ATN is false, any code of 8 bits or less understood by both talker and listener(s) may be used.

The other control lines are IFC, REN, SRQ, EOI.

IFC (interface clear) places the interface system in a known quiescent state.

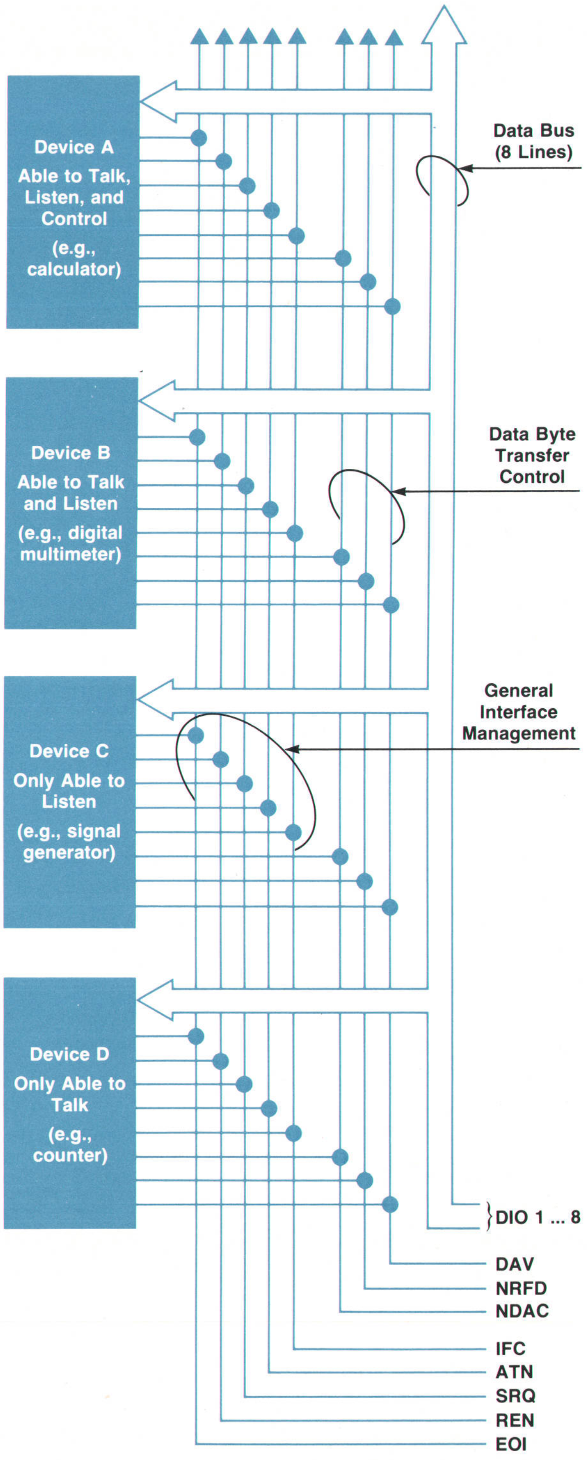
REN (remote enable) is used with other coded messages to select either local or remote control of each device.

Any active device can set the SRQ (service request) line true. This indicates to the controller that some device on the bus wants attention, say a counter that has just completed a time-interval measurement and wants to transmit the reading to a printer.

EOI (end or identify) is used by a device to indicate the end of a multiple-byte transfer sequence. When a controller sets both the ATN and EOI lines true, each device capable of a parallel poll indicates its current status on the DIO line assigned to it.

In the interest of cost-effectiveness it is not necessary for every device to be capable of responding to all the lines. Each can be designed to respond only to those lines that are pertinent to its function on the bus.

*American Standard Code for Information Interchange



To ensure a high degree of compatibility among products that are independently designed and manufactured at HP in widely scattered locations, the HP Interface Bus goes much farther than previous interface definitions in its scope and content. It provides Hewlett-Packard design engineers with the tools needed to interconnect a wide range of products from which systems can be configured with a minimal amount of additional engineering. Although it facilitates the assembly of systems, it does not guarantee the assembly of "instant" systems. Configuring a complete operating system demands detailed attention to all the device-dependent characteristics beyond the scope of the HP Interface Bus definition.

### What's New Since '72

The HP Interface Bus was first described in the October 1972 issue of the Hewlett-Packard Journal. Since that time, several aspects of the interface definition have been refined to make it more useful without compromising the original objectives. Typical of these refinements are the following areas of change:

- Signal line name changes made in response to the needs of international standardization, e.g. the MRE (multiple response enable) line is now called ATN (attention).
- Address extension (optional) to two bytes to permit a maximum of 961 talk and 961 listen addresses rather than 31 each.
- Physical extension of the maximum total transmission path to 20 meters, rather than 15.
- Enhancement of the service request protocol.
- Refinement of the parallel-poll capability.
- Addition of the EOI (end or identify) signal line.
- Addition of more addressed commands, e.g. DEVICE CLEAR.
- Specification of control shift capabilities.
- Remote-local protocol upgrade.

It is not the purpose of this article to describe these changes in detail but to alert the reader to the nature of the changes. In general, the bus structure remains basically as originally described. (More details about the bus are included in the operating manuals of some bus-compatible products).

### Relationship to Proposed Standards

Interest in an international interface standard applicable to programmable measuring apparatus has grown substantially during the past few years, paralleling the growing need for instrumentation systems In addition, there is an increasing desire to configure these systems from products made by different manufacturers. European organizations, particularly in Germany, were instrumental in initiating the standardization effort.

In mid-1972, Hewlett-Packard began to participate in various national and international standardization bodies to help develop a suitable interface standard. After initial goals were established by the U.S. Advisory Committee, the techniques used by the HP Interface Bus were adopted as an appropriate starting point for a draft document. An initial draft was written, evaluated by the Committee, and submitted as the U.S. proposal to an IEC (International Electrotechnical Commission) Working Group in the fall of 1972. Since then, the interface definition has undergone a number of minor changes to accommodate various needs at the international level.

In September 1974, the parent technical committee IEC TC66 approved the latest draft document for a formal ballot among the member nations of the IEC. The final results of the ballot will not be known until the end of 1975. Concurrently, a similar draft document is being evaluated as a potential IEEE Standard. The present definition of the Hewlett-Packard Interface Bus is compatible with the current IEC and IEEE draft documents.

It would be presumptuous for the Hewlett-Packard Company to forecast the eventual outcome of the draft document ballot, but it is worth pointing out that the widespread interest in this particular interface system outside of HP suggests that it satisfies many interface needs, that it simplifies the interface challenge for designers, manufacturers, and users alike, and that it does make a significant contribution toward providing more versatile and lower cost instrumentation systems.

## HP Interface Bus Specification Summary

| | |
|---|---|
| **Interconnected Devices:** | Up to 15 maximum on one contiguous bus. |
| **Interconnection Path:** | Star or linear bus network up to 20 meters total transmission path length. |
| **Signal Lines:** | Sixteen active total; 8 data lines and 8 lines for critical control and status messages. |
| **Message Transfer Scheme:** | Byte-serial, bit-parallel, asynchronous data transfer using interlocked three-wire handshake technique. |
| **Maximum Data Rate:** | One megabyte per second over limited distances; 250-500 kilobytes per second typical over full transmission path. |
| **Address Capability:** | Primary addresses, 31 Talk and 31 Listen; secondary (2-byte) addresses, 961 Talk and 961 Listen. There can be a maximum of 1 Talker and up to 14 Listeners at a time. |
| **Control Shift:** | In systems with more than one controller, only one can be active at a time. The currently active controller can pass control to one of the others. Only the controller designated as system controller can assume control. |
| **Interface Circuits:** | Driver and Receiver circuits TTL-compatible. |

# Putting Together Instrumentation Systems at Minimum Cost

*Instrumentation systems that do useful work can be assembled around the HP Interface Bus at costs in the $15k to $25k range. Here is an approach to assembling such systems with a minimum amount of engineering time.*

**by David W. Ricci and Peter S. Stone**

INSTRUMENTATION SYSTEMS CAN NOW be applied to a wide range of applications where system solutions were previously not justifiable on an economic basis. This is the result of recent developments that are making systems easier to design, build, and use, and thus cost-effective for many small-scale, low-volume measurements

In this article, we would like to discuss some of the techniques of assembling instrumentation systems based on the HP Interface Bus for use in various kinds of measurements. Some of the tasks for which these systems are particularly well suited are: (1) multiple or often-repeated measurements; (2) measurements needing real-time data reduction and/or decision making; (3) stimulus-response measurements; and (4) measurements requiring repeatability and accuracy.

### Why a System

The question, "do I need a system" has no clear-cut answer but must be based on an engineering evaluation of benefits versus costs. There are many benefits in using a system rather than a manual operation, some of which are:

- More consistent results in repeated measurements—a system is not subject to operator fatigue.
- Greater throughput because systems are generally faster.
- More thorough testing because system speed allows many more parameters to be measured in a shorter time.
- Results expressed in appropriate units since many systems controllers are capable of on-line data manipulation. A measurement of a thermistor's resistance, for example, may be converted directly to temperature.
- Greater accuracy; system errors can be measured automatically, stored, and accounted for in results.
- "Adaptive" data acquisition; a system can be programmed to branch to other measurements to help pinpoint the problem when it senses an abnormal condition.

The principal reason for *not* using a system is cost, not only the cost of the individual instruments used, but also the cost of special hardware needed, such as test fixtures, and the cost of preparing and debugging the software. It is possible that a thorough investigation of alternative ways of doing a job may point to an approach that can do the job reasonably well at less cost than a system. A system is really only a tool with which to solve a problem and regardless of how powerful the tool may be, it is nevertheless advantageous to select the right tool for the job.

### A New System Technology

The engineering costs of putting a system together, however, have been reduced significantly by three recent developments: (1) the HP Interface Bus; (2) the growing number of "smart" instruments with internal microprocessors; and (3) the advent of highly agile, "friendly" controllers that have a high degree of operator interaction.

The HP Interface Bus (HP-IB) has been the prime energizer in making systems a more attractive alternative. Its direct impact has been to simplify or eliminate many of the steps involved in system design and implementation. Its indirect influence has been as a catalyst during the design of new instruments to make them more useful in systems applications—more thought now goes into the design of a laboratory bench instrument in terms of its potential for systems applications. It has also sparked the development of a number of useful system accessories (see page 12).

The new "smart" instruments make it easier to apply the accuracy and versatility of the lab bench instrument to a system environment. Previously, lab instruments were seldom adaptable to systems work so special purpose system components of limited ability had to be used. The wider use of digital techniques made feasible by advancing semiconductor technology has made it easier to include the interface functions within an instrument so it can work more effectively in a system. It has also given the lab bench instrument the ability to process data and execute more complicated measurement algorithms, thus relieving some of the burden placed on the controller and the interfacing. It is not necessary to program each discrete action for these instruments—the interface now needs to handle only processed data and programming instructions that occur at a relatively low rate.

Another component contributing to the rise of the new system technology is the highly agile programmable calculator. These provide a "friendly" controller, useful for simple to moderately complex systems, with a high degree of operator interaction that greatly simplifies program generation and debugging.

Another benefit of the HP-IB capability is that a minisystem can be assembled for a one-time test and, once the test is performed, disassembled to allow return of the components to normal bench use—it is not necessary to have a lot of hardware sitting around idle between systems. These "one-shot" systems are usually put together in the engineering lab to evaluate or characterize certain devices and usually use a calculator as the controller because of the ease of developing programs. Engineers who have had experience in assembling these systems are able to plan, configure, write programs for and get results from a new system in only two to five days.

The interface bus also makes it easier to service a system. Operation of each device can be verified by testing it alone with the controller; the others are removed simply by disconnecting the cables.

## Recreatable Automatic Systems for the Lab

In the course of doing their own work, HP engineers have assembled a number of minisystems to quickly perform tests in the lab that otherwise would require considerable time to do manually. These are systems that are assembled around the HP Interface Bus for specific tests and then disassembled or reconfigured for others. Because of their wide applicability, many of these "one-shot" systems have been documented in a series of application notes (series 174-0) to help others assemble similar systems with minimum loss of time.

Typical of these is a system for measuring the transfer characteristic of voltage-controlled oscillators. It uses the Model 59303A Digital-to-Analog Converter (page 12) to derive a dc control voltage for the oscillator in response to calculator commands, a counter to measure the oscillator frequency, and a plotter to trace graphs of voltage versus frequency.

After the program is keyed into the calculator, all the engineer has to do is enter the voltage range and the voltage step size and then press RUN PROGRAM. The system then traces a plot in about 10 seconds.

The application note (AN 174-1) describes the equipment needed and how it is connected, and it gives the program listing. Other application notes in the series describe measurements of non-linearity in VCO's, short-term stability of oscillators, FM peak-to-peak deviation, and determination of probability densities.

### First Steps

Although the advent of intelligent instruments and controllers on a standardized interface has brought the systems approach within the reach of a much wider range of users and applications, it has not altered the fundamental process of designing and building a system. All the considerations that go into building a system still exist and must be evaluated. We propose the following procedure:

1. Define the problem
2. Select the instruments
3. Select the controller
4. Interface the devices
5. Integrate the system
6. Write utility software
7. Write applications software
8. Document the system.

Although this appears to be a step-by-step procedure, the design process is not that orderly—there is a good deal of iteration back and forth between steps. The reader should also be cautioned that this is not a magic formula that guarantees instant success. Use of the HP Interface Bus has not eliminated all the pitfalls—it just makes it easier to cope with them. This list merely outlines the considerations that must be evaluated.

### Defining the Problem

Defining the problem in terms of the results to be achieved is the most critical step. Without a clear defination at this point, it is difficult to make good decisions throughout the process, and subsequent effort may be wasted in backtracking. In fact, one cannot really determine whether or not a system approach is the best way until a precise statement of requirements has been established.

An instrumentation system can be thought of as an instrument in its own right, but the measurements it performs are generally broader and more complex than any single instrument can perform. Unlike a purchased instrument, however, the instrument system is partially designed by the ultimate user. The system designer must thoroughly understand the measurements he is trying to make, the trade-offs involved, and the techniques required to get the desired information. As with instrument design, failure to recognize the real needs will often get the system designer locked into a technique that may not be the best one for solving the problem. One of the biggest pitfalls in system design is trying to fit the solution to the problem, rather than the other way around.

How does one go about defining the problem? The best way is to make clear statements of the results to be achieved. This should ultimately result in a fairly detailed list of basic requirements and system features, stated as objectives. The trick here is to state the objectives in terms of results so a particular solution is not automatically indicated.

Objectives should be classified into two groups: Musts and Wants. Musts distinguish those requirements that are absolutely necessary; Wants are those that are desirable, but expendable. Want objectives may be weighted in importance to help in making trade-offs.

After the list of must and want objectives is established, a decision can be made as to whether or not a system represents the optimum approach. A system is indicated if it can do the job better, faster, more accurately, more economically or a combination of these. Other factors to consider are consistency of results and the need for data reduction.

### Selecting the Instruments

Selection of the proper instruments is not very difficult since measurement needs are generally quite specific once the problem is well defined. Care must be taken, however, to insure interfacing compatibility. Although the HP-IB solves the mechanical, electrical, and functional compatibility problems of interfacing, there are various operational differences. Each instrument generally has a different set of programming commands and/or data output formats. Failure to understand the syntax needed for each instrument can sometimes cause readings to be taken at the wrong time, cause a controller to interpret data incorrectly, prevent instruments from triggering when they're expected to, and so on.

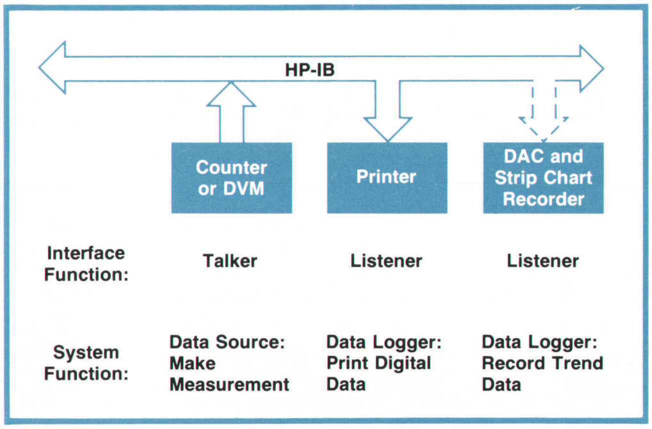


**Fig. 1.** *Elemental systems has one measurement device supplying data to one or more recording devices.*

At some point, the system designer may be faced with making a choice between using a bench instrument, either already designed for systems use or adaptable to it through various bus-compatible accessories such as code or D-to-A converters, or using a system component (a stimulus or measurement module designed specifically for systems use but not usable outside a systems environment). Although a systems component may be more cost-effective, it often lacks the high-performance capabilities of the lab-bench instrument. Besides, the lab-bench instrument is capable of manual operation as well as remote control, useful in debugging a new system and in system maintenance. Manual operation is also useful for diagnosing problems in a unit under test that is found to be faulty.

### Selecting the Controller

Traditionally, a system controller is a device that controls all the other devices in the system, performing such tasks as programming instrument modes, collecting and processing data, and so on. The HP-IB, however, defines a controller as the device that manages the operation and flow of data on the bus, a subset of the operations performed by a system controller. It is important to understand that an HP-IB controller does not necessarily program instruments or process data, although it may perform these functions also.

The various levels of controller complexity that are possible with the HP-IB are shown in the drawings. An important feature of the HP-IB is that the controller can be chosen independently of the instruments, enabling instruments assembled into a system at one level of system complexity to be operated at another level without changing the interface (provided the instruments' capabilities are adequate for a higher level).

The simplest configuration is shown in Fig. 1. This has a single measurement device, such as a counter or a DVM, outputting its data to a printer and, optionally, to a strip-chart recorder. The measurement device is operated in the "talk only" mode while the printer and/or strip-chart recorder are in the "listen only" mode. This means that whenever the measuring device places data on the bus, the listeners accept it without being addressed. There is no separate controller—the measurement rate is established by the measuring device.

The next level of complexity is shown in Fig. 2. Here a scanner functions as a simple controller. It addresses the talkers one at a time in sequence, and each transmits its data to the printer.

Both of the above examples are concerned primarily with data logging. No programming information is placed on the bus, so instrument functions and ranges must be set manually.

By replacing the scanner with a more complex controller that can address devices to talk and listen and that can send programming codes, the same collection of instruments can perform a wider variety of tasks, especially with the addition of a stimulus instrument. The higher-level programming can be done with a card or tape reader as the controller.

Once arriving at a stimulus/response situation,

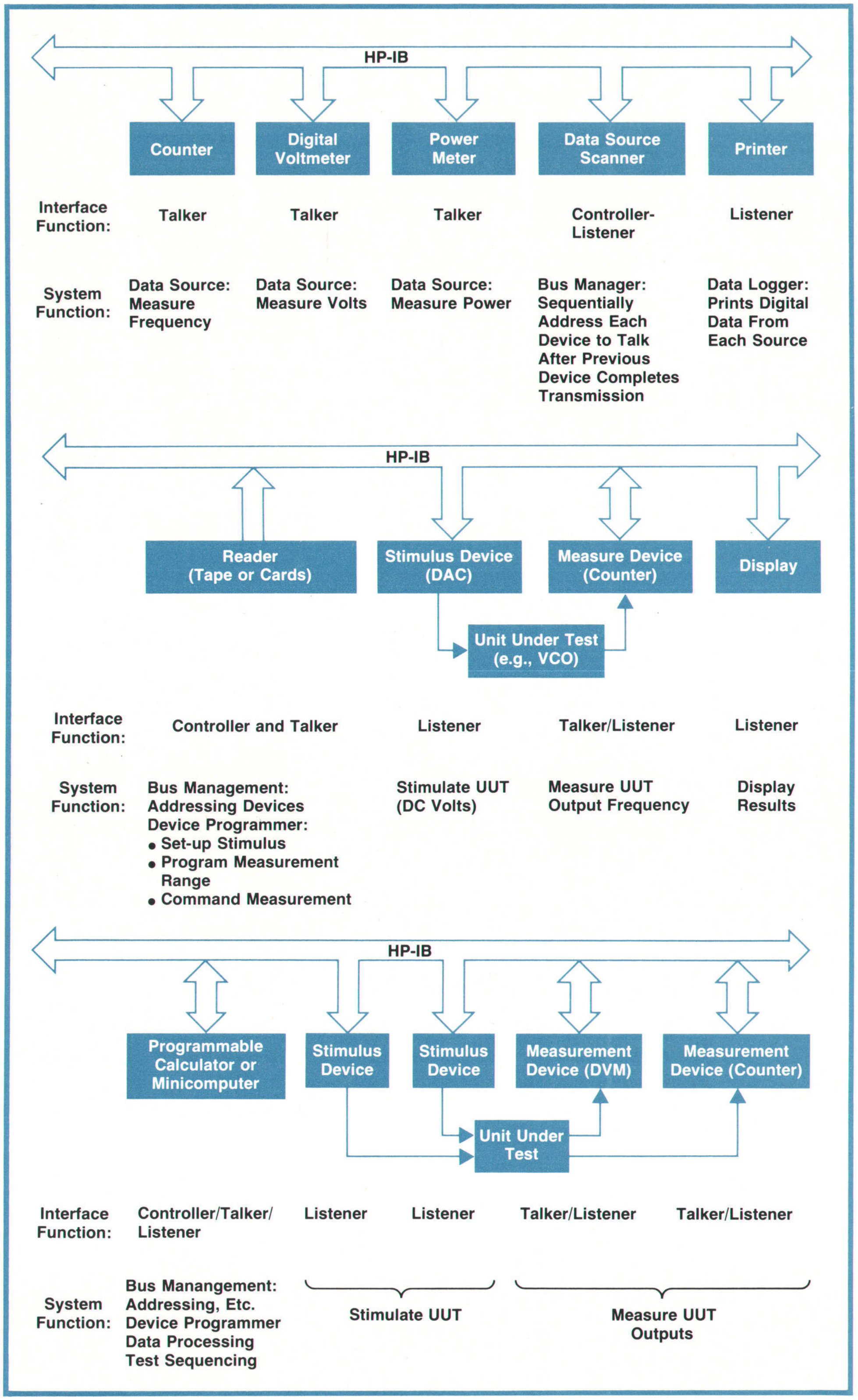


**Fig. 2.** *Addition of a scanner enables several measurement devices to take turns supplying data to a recording device.*

**Fig. 3.** *A tape or card reader as a controller enables ranges and functions to be programmed.*

**Fig. 4.** *With a calculator or computer as a controller, automatic date manipulation and decision-making can be included in the test program.*

## Developing a One-of-a-Kind Automatic Test System

One of the first systems built around the HP Interface Bus was a production test system for the HP Model 5340A Microwave Counter.

The goal was to shorten test time. During the early production of this instrument it became apparent that the limiting factor in the quantity that could be produced was the capacity of the test station, several hours being required to test each instrument. The most economical way to break this bottleneck, it was decided, would be to automate the procedure.

The problem could be stated very simply: to completely verify the operation of the Model 5340A and check all its specifications it must be tested at many frequencies between 10 Hz and 18 GHz at several power levels between −45 dBm and +10 dBm, and in all operating modes. This was a "must" objective. A "want" objective was to make the test unattended so the test technician could spend his time troubleshooting units that failed the test.

Selecting the instruments for this system posed a problem. At that time, there were no HP-IB-compatible signal sources that operated above 1.3 GHz. So, for the higher frequencies HP 8690-series sweepers were used, monitored with a 5340A Counter to give precise frequency information. A programmable attenuator and a 432C Power Meter provided control of the power level.

The lower range of frequencies was covered by a 3320A Synthesizer for the 10 Hz-to-13 MHz range and an 8660A Synthesizer for the 13-to-1300 MHz range, both HP-IB-compatible.

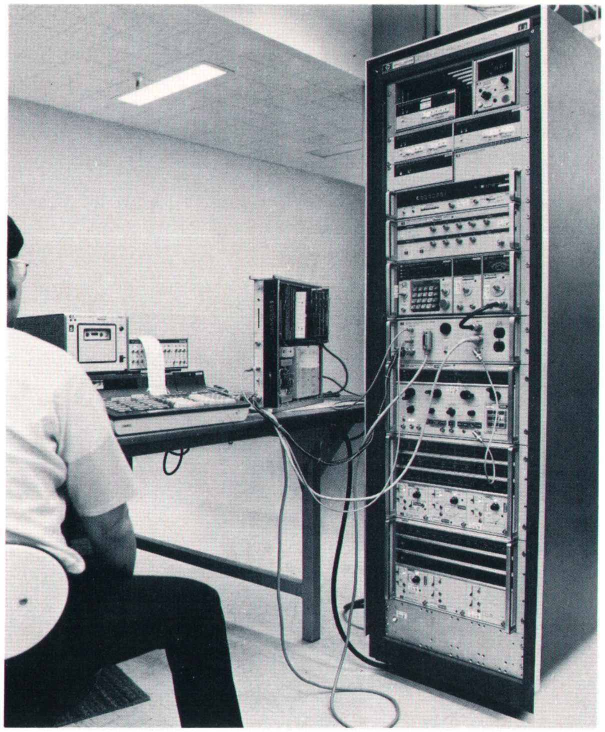
Selection of a controller was dictated by the need for programmability and the need for some computation capability, such as finding the logarithm of power readings and comparing readings to determine whether or not a reading is within tolerance. The Model 9820A Calculator was selected for this task but the desire to make the system run unattended required the addition of a tape cassette memory to accommodate long programs (the Model 9821A with its built-in cassette memory was not available at the time).

There was no problem, of course, interfacing the HP-IB-compatible instruments. The programmable attenuator was easily interfaced by way of the Model 59306A Relay Actuator (page 12).

Interfacing the non-HP-IB instruments, however, required some effort. This was accomplished by use of Model 59301A ASCII-to-Parallel Converters (page 12) controlling versions of the sweepers and power meter that had been adapted for computer control through a BCD interface. Additional circuits had to be designed, however, to match the logic levels of the sweeper interface to those of the ASCII-to-Parallel Converter.

Integrating the system largely involved careful consideration of how to run the RF cables to minimize VSWR and resultant losses.

There was no problem with the utility software since all the instruments were controlled through the interface bus one way or another. Writing the applications software was straightforward. It involved test strategy and some diagnostic programs that prepared failure reports to help the test technician locate troubles in the instrument under test.



After the system was assembled, debugged, and running, another "want" objective came to light. This was a desire to give the test technician some indication that a test was completed or otherwise stopped. Because the system did not use all six relays in the Model 59306A Relay Actuator, one was available for ringing a chime. A program loop was written to ring the chime five times whenever the test program was stopped.

A complete test of the Model 5340A Counter is now completed within 40 minutes, and this is an unattended test.

Because of the success of this test system, there was little argument about whether or not to use an automatic test system for the next high-performance instrument to come along, the Model 5345A. Testing requirements for this instrument were similar to the 5340A, but the frequency range goes only to 500 MHz. Thus, all but one of the instruments needed were bus-compatible (the exception was a pulse generator, but since it would be used in one mode only it did not require programmability). As a result, the system was assembled and up and running in less than 20% of the time that had been taken for the 5340A test system.

The 5345A test system was developed by Tom Coates. Al Foster designed the automatic test system for the 5340A.

some sort of feedback to the operator may be needed to indicate whether or not test results are within specified limits. Such a situation is diagrammed in Fig. 3.

As soon as this level of complexity is reached, however, there will likely be other requirements for data manipulation or automatic decision making. Thus, the use of a programmable calculator or a computer as the controller is indicated, as shown in Fig. 4. Here, all the potential advantages of programming instruments and accessories can be brought to bear on the problem.

The question of which level of controller to use is largely answered by the level of decision making and data manipulation required. The decision to use a lower-level controller is not a binding one, however, since a system can be upgraded to a higher level, assuming that the other system components have the necessary capabilities. Upgrading is simplified by the fact that no changes in the interface are required to do so.

The choice between a calculator and a computer is not so straightforward. For systems of simple to medium complexity we, as design engineers, have found the programmable calculator to be a powerful controller that is especially easy to use for program generation and editing. Where a great deal of on-line storage may be needed, a computer is indicated. The computer also offers more flexibility in terms of language and software operating systems, and it offers the potential for higher speed.

### Interfacing the Devices

Interfacing used to be the major problem in assembling a system—each device required a separate piece of interface hardware and, very often, a separate software driver as well.

Now, if all the components selected for a system are compatible with the HP Interface Bus, the hardware interfacing is already done. Each instrument has its own I/O facilities for communicating on the bus, and they are linked together simply by connecting them with passive cables.

If a required instrument function is not available with the HP-IB , then one has to decide if it is possible to obtain the desired function by using some of the bus accessories (page 12) to drive a standard instrument. If that is not possible, then the engineering effort to interface the instrument must be evaluated. As a greater variety of instruments are developed for the HP-IB, this should become less and less of a problem.

### Integrating the System

Integrating the system is simply a matter of assigning addresses to all the devices and connecting them with standard HP-IB cables. Again, if all the system components are bus-compatible, there is no real problem.

One aspect of system design that often is overlooked until too late is that of "fixturing"—connecting the system to the unit under test or to measurement points. This may involve the switching of low-level analog signals with resulting cross-talk and accuracy problems and can require extensive development time, especially in fully automatic systems. This problem is highly application dependent, so it is difficult to characterize generally other than it must be considered in the overall system design. Failure to do so may negate the effectiveness of the remainder of the effort.

### Writing Utility Software

At this point, a means of controlling the communications to and from each device in the system must be developed. Usually, this is in the form of software. For convenience we have divided the software into two parts: the utility software, which is the instrument-dependent driver software for handling the I/O requirements of each device in the system, and the applications software which is concerned with the measurement algorithms. Applications software is largely device independent.

The utility software is greatly simplified by the nature of the HP-IB. The HP-IB addressing structure allows all bus instruments to share common driver routines thus reducing the amount of specialized software that needs to be written. Because the HP-IB is basically a communications structure that does not require an understanding of content to function, the utility software can be developed to provide the communications to and from a device without regard for the particular device's characteristics.

The simplest form of software for an HP-IB system would be that required of a card reader to manage the bus, i.e. send addresses and bus commands. Here, the binary code for each data line on the bus and for the ATN line must be marked on the card for each byte of information the card reader is to place on the bus. Where an HP 9800-series Calculator is to be used as the controller, this kind of detail is handled by a plug-in ROM block so the operator can control the devices through the higher-level language of the keyboard.

### Writing the Applications Software

The availability of desk-top calculators with their readily grasped program generation, editing, and debugging techniques combined with the standard communication techniques used on the HP Interface Bus and system-oriented bench instruments makes the generation of application software much easier than it has been. But, if any one step in building a system could be called the most significant, writing the applications software is it! This details when and how measurements are to be taken and how the raw measurement data is to be processed.

Too often a system builder underestimates the extent of the effort required to achieve desired results—there can be a large discrepancy between having the capability to make a measurement and actually making it. In stimulus/response testing, for example, applications software is heavily involved in test strategies—where to start, what sequence of tests, how many tests are needed, tests limits and guard banding, failure analysis, troubleshooting aids, etc., etc. Even with a packaged system, software development is still required to solve specific problems the user has in mind because applications are too diverse to lump into one universal package.

For very simple systems, such as one using a data source scanner as the controller, applications software is practically non-existent as the user sets all ranges and functions manually. But as soon as the controller requires the specification of program steps, then the applications software assumes major proportions.

The system designer should keep in mind who will use the software. We have found that with small-scale production test systems, the use of a programmable calculator with its easy-to-learn keyboard makes it possible for the test technician who runs the test to contribute valuable software improvements even though he may not have had prior programming experience. If the software-user interactions are planned properly, the program can allow the test technician to decide what course of action to take when a failure is encountered in the device under test—for example, he may want to loop on the failure while he probes with an oscilloscope or he may simply want an error message printed and the test continued. The point is, good software should be flexible; it should allow the system user to select reasonable variations of the procedure depending upon the particular conditions encountered during a test.

Good system software should also take into account the mistakes a human operator is likely to make when inputting system information. Many systems require a fairly large number of input parameters before the actual test or measurement cycle begins. The entire system is much more usable if the software is written to allow the user to correct isolated input errors without having to restart the entire program. In general, it is much more advantageous to tailor the software to human characteristics than to try to train the human to cater to the software.

## Documentation

Sometimes overlooked is the need to write down operating procedures so others who need to use the system can know what to do. System performance verification and servicing procedures also need to be worked out for maintaining the system. It often helps to also have verification procedures written for each individual device in the system as troubleshooting aids.

Often overlooked is the need to document the system configuration so it can be readily rebuilt if it should happen to be disassembled, or if the need should arise to duplicate it.

## Packaged Calculator-Based Measurement Systems

Because one user's requirements are generally not quite the same as another's, most applications of automatic systems are unique in one way or another.

Nevertheless, HP engineers have been able to identify a number of applications that use the same hardware, and very often some common software. Because of the hardware commonality, equipment for a number of these applications is now offered in packaged systems that have all the hardware complete and tested as a unit.

These are not "turnkey" systems, however, in the sense that they can be put to work as soon as installed—software needs to be prepared for each particular application. The package, though, does include useful subroutines and examples of small applications programs that help in getting the system "on the air". Very often, these programs can be modified for particular purposes.

The Model 3050B Automatic Data Acquisition System is an example of the kind of calculator-based system offered. This system includes the Model 3490A Digital Multimeter (1-$\mu$V resolution, 120-dB system common-mode rejection), the Model 3495A Scanner (see page 17), one of three calculators (Models 9820A, 9821A, or 9830A), an equipment rack and cabling.

The basic version measures dc volts, ac volts, and resistance through as many as 40 channels. Besides logging data, it can do simple go/no-go limit testing. However, it can also process data, such as compensate readings for transducer nonlinearities, convert readings to engineering units, and do statistcal analyses (determine average values and standard deviations, and do trend analyses).

Many peripherals, such as a plotter and a timing generator, are available to broaden the system's capabilities. Because it uses the HP Interface Bus, it is easily expandable.

Other available calculator-based systems include the Model 3045A Automatic Spectrum Analyzer that performs spectral analysis, distortion analysis, and wave analysis over a frequency range of 10 Hz to 13 MHz with selectivity as fine as 3 Hz, and the Model 3042A Network Analyzer that measures phase response along with amplitude response over a 50 Hz to 13 MHz frequency range.

Typical prices in the United States for basic systems are $21,950 for a Model 3042A Network Analyzer, $22,400 for a Model 3045A Automatic Spectrum Analyzer, and $14,100 for a Model 3050B Data Acquisition System with ten low-thermal channels.

# Filling in the Gaps—Modular Accessories for Instrument Systems

*These programmable modules provide such accessory functions as remote display, switching, digital-to-analog conversion, and measurement pacing and timing. They are useful both with single instruments and as components of automated systems.*

**by Steven E. Schultz and Charles R. Trimble**

EVERY DAY, ENGINEERS APPLY bench instruments such as counters, voltmeters, and synthesizers to the solutions of measurement problems. But very often something more than the basic instruments is needed to complete the job. This something more might be a signal switch, a digital-to-analog converter, a pacer, a relay actuator, or some other accessory.

This need for something more led to the development of a series of programmable modular accessories for instrument applications. The modules were designed with a dual purpose in mind: to work as accessories for stand-alone instruments, or as components of an HP Interface Bus-connected automated system where they fill the gaps in the system. A description of the modules presently available is given here to help the potential system builder envision how he might implement a solution to his measurement problem.

### Digital-to-Analog Converter

The Model 59303A Digital-to-Analog Converter accepts up to 15 ASCII-coded digits serially, stores them, and on receipt of the line-feed character produces an analog voltage within a range of ±10V equivalent to three consecutive digits selected from the string. Among other uses it can be used to convert the digital output of a counter or a DVM to an analog voltage for driving a strip-chart recorder or an X-Y plotter.

The new D-to-A Converter converts any three consecutive digits within the received character string to the equivalent analog voltage with an accuracy of 0.1%. The relationship between the output and the three digits ($D_1$, $D_2$, $D_3$) is as follows:

$$\text{Analog output voltage} = (D_1 \times 1\text{V} + D_2 \times 0.1\text{V} + D_3 \times 0.01\text{V}) \times \text{the polarity sign}$$

Ignoring decimal points, the converter selects three digits in any data format from fixed point to scientific notation. When under remote control, it can be programmed to select the digits from either of two data words sent in the same character string. This ability is useful when the measuring instrument has two outputs, such as phase and amplitude, transmitted on the interface bus. One 59303A can convert the information in the first number for a plot of phase versus frequency, and a second 59303A can convert the second number for a plot of amplitude.

A block diagram is shown in Fig. 1. A "listen-only" mode, selected by a rear-panel switch, causes the D-to-A Converter to respond to all inputs. It may thus be used directly to convert the output of any bus-compatible "talking" instrument without use of a bus controller. In the "addressable" mode, it responds to inputs only when addressed to do so by the system controller.

Other uses for the D-to-A converter are as a programmable voltage source for testing other devices (it can sink or source up to 10 mA), to program trigger levels on counters, and to program analog-controlled devices such as sweepers and voltage-controlled oscillators. A precision frequency source can be obtained by using a calculator to set a number into a D-to-A converter controlling a VCO, and a counter to report the VCO frequency back to the calculator. The calculator compares the counter reading to the selected number and then adjusts the digital input to the D-to-A converter to minimize the difference. Only a few iterations are required to reduce the difference to the required low level.

Three output formats are provided: −9.99 to +9.99V, 0 to +9.99V ignoring the sign of the input data or 0 to +9.99V offset where a zero input produces 5.00V out. This last mode is useful for avoiding jumps from full scale to zero when plotting

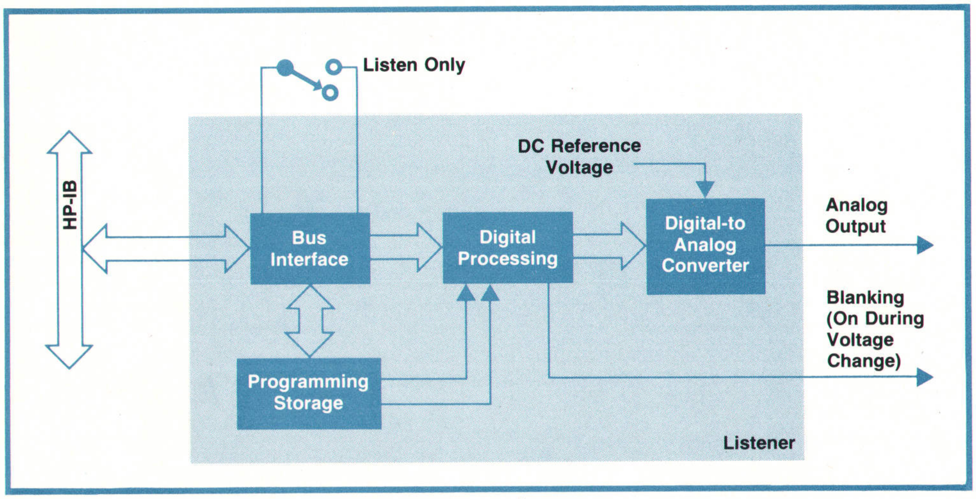


**Fig. 1.** *Model 59303A Digital-to-Analog Converter derives dc voltage equivalent to three consecutive digits out of a string received serially in ASCII code.*

bipolar inputs on a unipolar strip-chart recorder, as when plotting the output of a counter that is monitoring the long-term stability of a 10-MHz oscillator.

## A Timing Family

Two modules—a timing generator and a clock—were developed for control of time functions.

The Model 59308A Timing Generator functions either as a digital delay generator (timer) or as a precision time marker generator (pacer). A block diagram is shown in Fig. 2.

As a timer, the Generator counts down crystal-controlled 1-MHz pulses and, following the receipt of an input trigger, generates an output pulse when the selected number of 1-$\mu$s time increments has elapsed. It is, in effect, a programmable one-shot with a range of $1 \times 10^0$ $\mu$s to $999 \times 10^8$ $\mu$s. This mode is useful for programming a delay, for example to allow a power supply to slew to a new voltage level before it is used in a measurement.

The time delay is programmed through the HP Interface Bus or, for local control, it is set on front-panel switches. In either case, three digits and a power-of-ten multiplier are entered ($\mathrm{DDD} \times 10^{\mathrm{D}}$) in units of microseconds.

In the pacer mode, the generator produces a pulse every $\Delta t$ on and following the receipt of a trigger, where $\Delta t$ is the number set into the front-panel switches or programmed through the HP Interface Bus. In this mode it may be used as a precision sample rate generator, say, to trigger a voltmeter reading every 10 seconds.

It may also be used for time interval measurements. It has an internal six-digit (decimal) counter that totals the number of output pulses produced since a trigger was received. It could be pro-

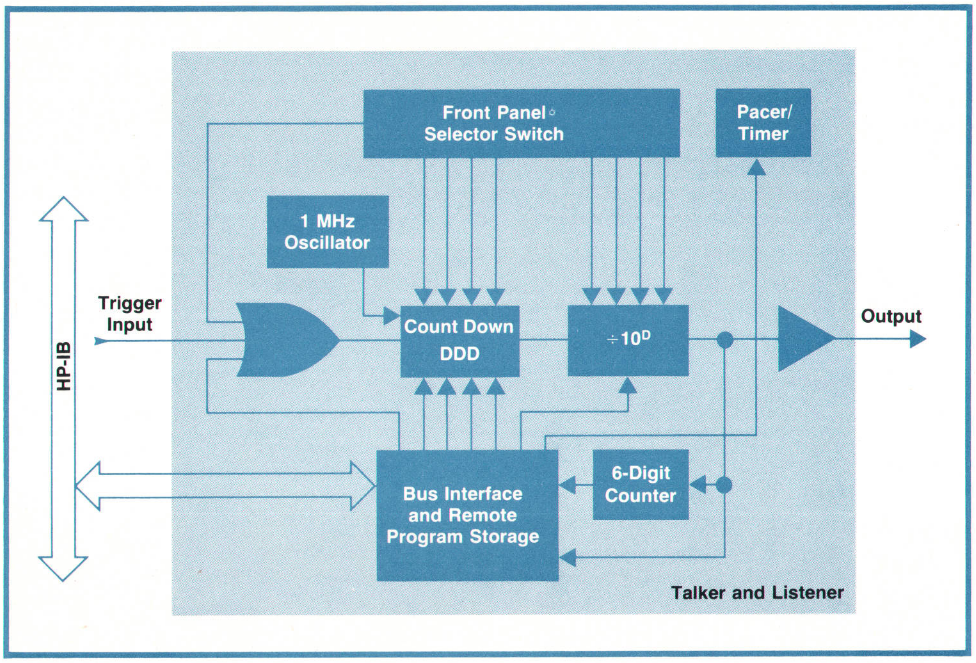


**Fig. 2.** *Model 59308A Timing Generator functions as a digital delay, a time mark generator, and a time-interval counter.*

# A Quiet, HP-IB Compatible Printer that Listens to Both ASCII and BCD

### by Hans-Jürg Nadig

The need for a low-cost, bus-compatible printer coupled with a desire for quieter operation has led to the development of a new printer, the Model 5150A (Fig. 1).

This printer is not limited to recording data in systems using the HP Interface Bus, however. It works as well with the older BCD instrument interface. It was designed with an internal data bus to give it a flexible, option-based architecture that provides the versatility needed for a wide variety of applications. The user can "design" his own printer according to his needs without paying for capabilities he doesn't need, and he can update it to a different configuration whenever he wishes.

Besides accepting the widely-used BCD data from a variety of instruments or ASCII-coded characters from the HP-IB or other ASCII-coded data sources, the new printer can also control the timing of data acquisition and direct the sequential acquisition of data from several sources. It can also print the time of day on the data record.

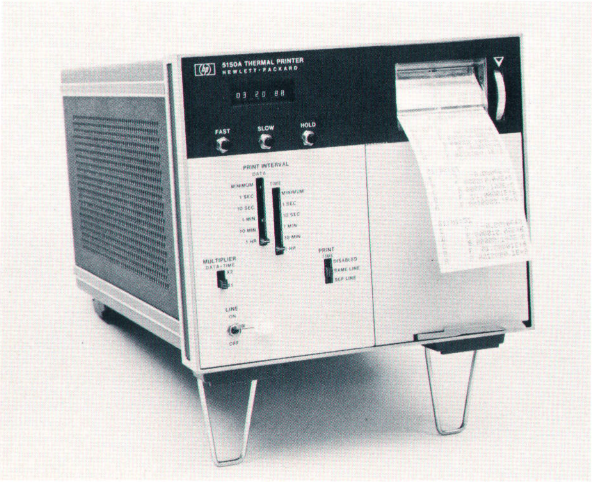


**Fig. 1.** *Model 5150A Thermal Printer.*

## Plug-In Options

In the interest of keeping the printer as simple as possible, the basic instrument contains only the print mechanism, a power supply, and internal control circuits (Fig. 2). There are no functions in the basic unit that are not needed by all options. As a consequence, one of the plug-in interface options must be installed for the printer to become a functional unit.

Four options are presently available: (1) an ASCII interface for communicating on the HP Interface Bus; (2) a BCD parallel 10-column input (two can be installed for 20-column print-out); (3) a scanner for controlling the HP Interface Bus, for sequentially addressing a number of instruments and printing their readings; and (4) a digital clock for printing time with the data and for controlling the rate of data acquisition (data output of the clock to the interface bus is not provided). Three of the options are on plug-in circuit cards that include the necessary external connectors. The clock option is mounted behind the front panel and does not occupy either of the two plug-in slots.

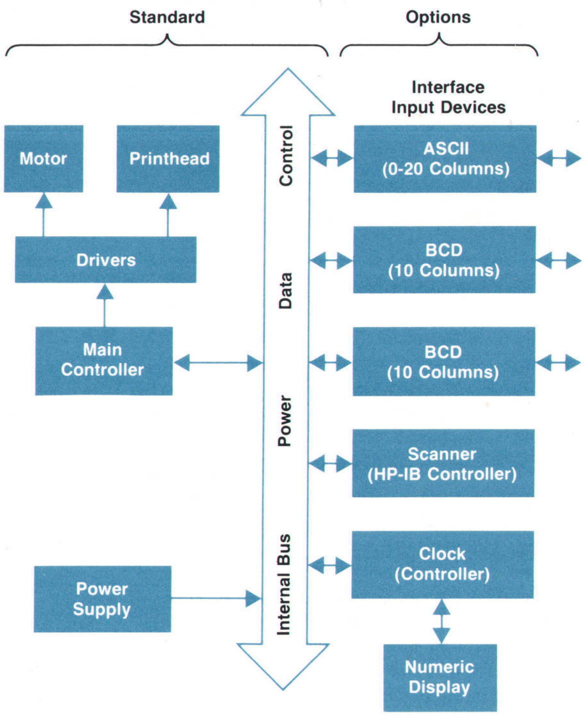


**Fig. 2.** *Basic organization of Thermal Printer.*

## More Than a Printer

In its simplest application, the new Model 5150A can do as earlier printers have done; log data from an instrument. When used with the HP Interface Bus and operated in the "listen-only" mode, it accepts all data appearing on the bus. When it senses the LF command, it prints the most recent 20 characters (or fewer if less than 20 have been received since the last LF command). In the "addressable" mode, it accepts data only when addressed.

The combination of the clock, scanner, and HP-IB options places the printer beyond the realm of a simple data logger. The clock can initiate a scan at intervals selected on the front-panel DATA PRINT INTERVAL control. The scanner will then address the lowest numbered instrument on the bus, wait for the instrument to send its data, print that data, then address the next instrument. The time required to complete a scan thus depends on the instrument response times.

The clock can initiate scans at intervals shorter than one second and as long as two hours(if a scan is still in progress when a clock trigger occurs, the scanner ignores the trigger). It can thus be used for short-term tests in the R and D lab, the quality assurance lab, and the production test stand, or it can be used for unattended monitoring over long periods of time, as in industrial processes or checking conformance to FCC regulations. When the scanner is used without the clock, the scan cycle repeats continuously.

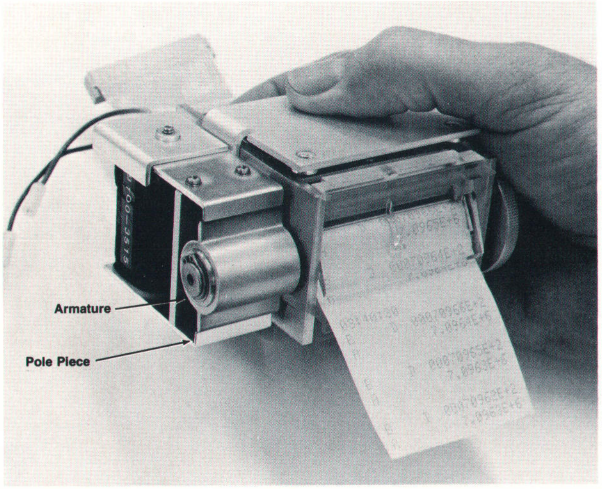


**Fig. 3.** *Print mechanism is uncomplicated.*

### Few Moving Parts

The new printer uses the thermal-print technology developed for the HP-9800-series Calculators*. Alphanumeric characters are printed on a 5 × 7 dot matrix as the heat-sensitive paper is stepped past the thermal print-head. With the ASCII Interface, it prints characters from the ASCII 64-character upper case printing set. With the BCD interface, the printer has a repertoire of 16 characters, normally the digits 0 through 9, +, −, V, A, R, and blank. Any other set of 16 characters may be printed by changing ROM's within the instrument. It is even possible to configure the printer to print entire words in response to a single BCD input. It prints up to 20 columns at a maximum rate of 3 lines per second.

In the interest of achieving long-term reliability at minimum cost, the paper drive uses a very simple stepping motor. The motor armature is a cylinder with two winged projections (see (Fig. 3). Whenever the field magnet is pulsed, the wings are drawn into alignment with the pole pieces. Between pulses, a spring (hidden behind the armature in the photo) rotates the armature about 5° backwards against a stop. The resulting oscillating movement of the armature drives the paper-drive shaft through an overrunning clutch. A data line is written in 7 increments with 3 more provided for interline spacing.

Drive pulses occur at a 30-Hz rate and because the incremental movement is very small, motor operation is barely audible. The motor is the only mechanical motion in the printer—there are no print wheels, hammers, or inking systems.

The paper drive was developed by Ron Jensen, who also contributed a large portion of the total design. Product design was by Bill Anson and Keith Leslie.

*D.B. Barney and J.R. Drehle, "A Quiet, Low-Cost, High-Speed Line Printer", Hewlett-Packard Journal, May 1973.

grammed, for example, to generate a pulse every millisecond (100E1) so the stored count would give a reading of the number of milliseconds since a trigger occurred. The stored count is output to the interface bus by addressing the generator to talk.

The timing generator functions either as part of an interface-bus-connected system, receiving and sending trigger indications through the bus or through rear-panel connectors, or in stand-alone applications using the rear-panel trigger input and output connectors. It could be used, for example, to establish a precision data rate in a voltmeter-printer system that does not have a controller.

Two timing generators can be used as a programmable pulse generator. One establishes the repetition rate and serves as a trigger generator for the second one. The second is operated in the square-wave mode, in which it generates a rectangular pulse equal to one-half the time value set into the front-panel switches, thereby establishing pulse width.

### The Clock

The Model 59309A ASCII Digital Clock gives absolute time in seconds, minutes, hours, days, and months. When connected to the HP interface bus and asked to talk, it outputs the time on the bus. A block diagram is shown in Fig. 3.

The Digital Clock is a precision instrument, using a 1-MHz quartz crystal resonator in its master oscillator. The aging rate of the crystal is 5 parts in $10^6$ per year. The clock can also be driven by an external frequency standard of 1, 5, or 10 MHz.

It has other features that make it more than an ordinary digital clock. With a standard 9-volt battery installed for standby power, it becomes immune to powerline transients and it can operate on the battery for as long as a full day when there is a power-line interruption (the display will be turned off, however). It can operate with any other 8-10V dc power source through a rear-panel connector (it draws 2 mA at 8V with the display off). A companion unit, Model K10-59992 Standby Power Supply provides up to a year of standby power using size D flashlight cells.

Another useful feature is an internal memory that stores the time on command for later output. This would be used to store the time of a voltmeter reading at the instant the reading is taken, for later printout.

The clock can be set by codes sent on the HP interface bus. It is thus possible to create a subroutine that automatically sets the clock on system start-up. It can also be set manually with switches that are behind a front-panel lift-up door.

### A Code Converter

Pre HP-IB instruments such as HP's 5050A/B Digital Recorder and 580A Digital-to-Analog Converter can be operated on the HP Interface Bus using the Model 59301A ASCII-to-Parallel Converter. This "interface to the interface" accepts ASCII-coded data from the HP interface bus and converts it to the BCD code used by these and many other instruments. It is thus possible to obtain a 20 line/second hard-copy log from an HP-IB system using the Model 5050B Printer with the ASCII-to-Parallel Converter.

ASCII-coded characters are first converted to the equivalent 4-bit BCD or hexadecimal equivalent, then fed to four 16-bit shift registers. The contents of the shift registers are output in parallel on two rear-panel 50-pin connectors. On receipt of the line-feed character, a "strobe" command is sent to the parallel interface to indicate that the data is complete. Up to 16 ASCII-coded characters can be handled per byte.

The converter has a "listen only" mode that enables its use in controllerless systems. In the "addressable" mode, it processes characters only when addressed by a system controller.

### Supplemental Display

For display of data transported on the interface bus, the Model 59304A Numeric Display includes a memory for temporary storage and display of up to 12 digits and a decimal point. This module is useful for storage and display of intermediate results, for example so the program does not have to be slowed for display refresh in a calculator-based system.

The numeric display has a "listen only" mode so it can be used as a remote display for a measuring instrument. This is useful in RF or microwave systems where the actual measurement must be performed at an inconvenient place—it is much easier to transmit the data digitally on the interface bus then it is to reroute signal-carrying cables.

### Automatic Control

The Model 59306A Relay Actuator has six form-C relays for control of equipment ranging from microwave switches to environmental chambers. The relays have both normally-open and normally-closed

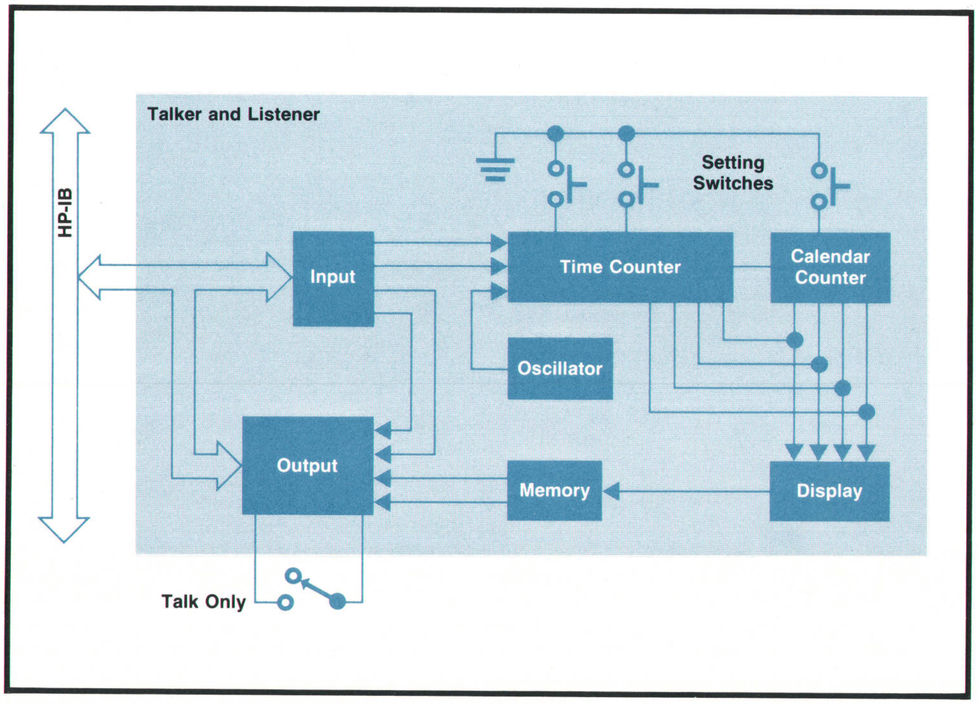


**Fig. 3.** *Model 59309A ASCII Digital Clock is settable by front-panel and remote control.*

# A Multifunction Scanner for Calculator-Based Data Acquisition Systems

### by David L. Wolpert

In the usual data acquisition system, a scanner switches several inputs one by one to a single instrument for measurement and recording.

The new Hewlett-Packard Model 3495A Scanner does this and much more. For example, it can serve as a programmable switch for actuating external processes or for distributing power and/or signals. It can also connect more than one channel at a time so it is able to make the multiple connections needed for four-wire resistance or floating bridge measurements. Where there may be more than one measuring and/or stimulus channel to be connected to a multiport device, the new scanner can also be configured to make the multiple switch closures needed for matrix switching.

The scanner is designed to work with the HP Interface Bus in calculator-controlled systems. It is addressable and once addressed to listen, it accepts and stores channel addresses until it receives the EXECUTE command, which then causes the indicated channels to open or close. It switches up to 40 channels and up to five scanners can be connected to the bus to give break-before-make operation of up to 200 channels.

Channel switches are mounted on plug-in circuit modules, 10 channels to a module. The mainframe holds up to four modules in any combination desired and is easily reconfigured in the field to meet the needs of new measurement situations.

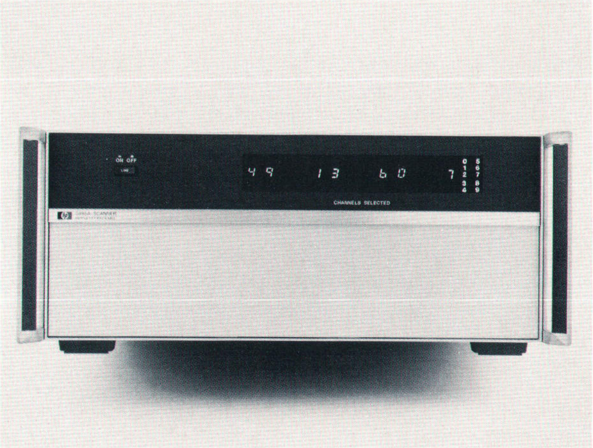
Two types of modules have been designed. One, known as the low-thermal assembly, was designed for minimum offset voltages ($<2\ \mu V$) and high impedance ($>10^{10}\Omega$ isolation) for use with thermocouples and other low-level transducers as well as high-level signals. It has three reed switches per channel. One switch in each channel connects the circuit guard and is programmed to close before and open after the HI and LO signal switches.

Only one low-thermal channel may be closed at a time on any one module but channels on adjacent modules may be closed simultaneously, making it possible to have up to four simultaneous closures if four modules are installed. Jumpered connections on each module enable channels on two or more modules to respond to the same channel address for simultaneous switching of channels.

The maximum terminal-to-terminal voltage permitted is 200V. Switching time is less than 10 ms with hardware insured break-before-make switching.

The other type of circuit module is known as the actuator assembly. Each of these modules has 10 general-purpose, double-pole, single-throw, armature-type relays with contacts rated at 100 V max, 2 A max. Any combination of channels on one of these boards may be closed simultaneously.

This card has a single unswitched guard for all 10 channels. Switching time is less than 40 ms, giving a maximum closure rate of about 30 channels per second.



# Instruments for Use in HP Interface Bus Connected Systems

In addition to the instruments described in this issue, there are a number of Hewlett-Packard instruments that operate on the HP Interface Bus when equipped with the appropriate options. These include the following:

**Signal Sources**

- 3320B Frequency Synthesizer, 0.01 Hz to 13 MHz
- 3330A/B Automatic Synthesizer/Sweeper, 0.1 Hz to 13 MHz
- 8660A/B Synthesized Signal Generator, 10 kHz to 1.3 GHz
- 8016A Word Generator, 9 outputs, 32 bits each, 0.5 Hz to 50 MHz

**Measuring Instruments**

- 3490A Digital Multimeter, dc volts, ac volts, ohms; also ratio and sample/hold options
- 5340A Automatic Frequency Counter, 10 Hz to 18 GHz
- 5341A Automatic Frequency Counter, 10 Hz to 4.5 GHz
- 5345A Plug-in Electronic Counter, to 500 MHz direct and higher with plug-ins; time interval down to 2 ns

**Calculators**

- 9820A Algebraic Language Calculator, magnetic card programming
- 9821A Algebraic Language Calculator, tape cassette programming
- 9830A BASIC Language Calculator, typewriter-style keyboard

contacts and can switch up to 25V at 0.5A.

Signal switching is performed by the Model 59307A VHF Switch. This module has two four-position, bidirectional switches, consisting of miniature relays that maintain a 50-ohm characteristic impedance for the signal path.

Designed to maintain fast pulse transition times, the VHF switch is useful for selecting trigger inputs for time interval measurements, among other uses (signal path lengths are matched within 10 ps). It is also useful for routing the outputs of signal sources to various points in a measurement system.

Besides being actuated by instructions received over the HP interface bus, both the VHF switch and the relay actuator may be activated by front-panel pushbuttons. Local control facilitates system set-up.

#### Acknowledgments

Special credit is due Eric Havstad and Carl Spalding for the product design. John Dukes and Jim Sorden fostered initial concepts of the modules with Jane Evans providing valuable marketing support.

## Minimal Cost Measuring Instruments for Systems Use

### by Gary D. Sasaki and Lawrence P. Johnson

Although the HP Interface Bus technology makes it possible to use very sophisticated laboratory instruments in automatic measuring systems, many requirements can be fulfilled with more modest instrumentation.

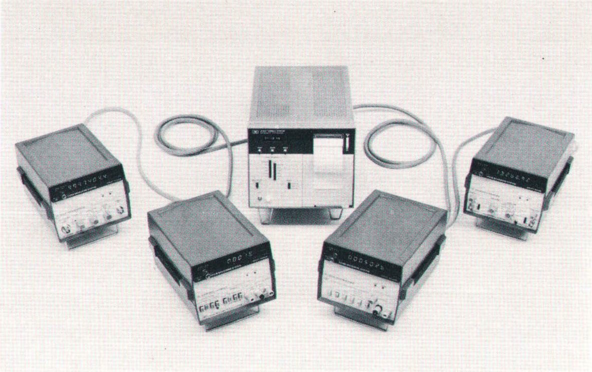
The HP 5300-Measuring System (counters, timers, and a multimeter) has the potential for providing a wide variety of capabilities for low-cost automatic systems. This system consists of mainframes that have basic counting capability, an LED display, and a time base. Signal conditioning circuits are in functional modules that mate with a mainframe to form a complete instrument. The transfer of signals between the functional module and the mainframe by way of a multi-pin connector provides an ideal means to gain access to measurement data for use on the HP Interface Bus.

Accordingly, a new "snap-between" module, the Model 5312A ASCII Interface, was designed to make the measurement data available in ASCII code to the HP Interface Bus. This module works with the Model 5300B Mainframe to make any of the eight presently available plug-on functional modules a measurement entry port for systems using the HP Interface bus (the Model 5300A Mainframe does not make certain status signals available to the module connector so cannot be used with the ASCII Interface).

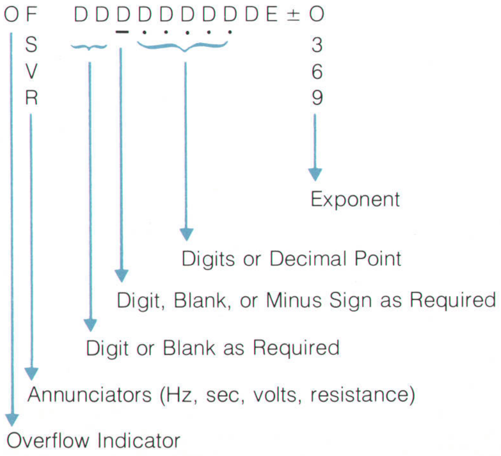
The measurement capabilities made available through various plug-on modules by this arrangement are frequency (up to 1.1 GHz), period, time interval, total count, frequency ratio, ac volts, dc volts, and resistance. Besides amplifiers and attenuators, the functional modules may have trigger circuits and the logic for making time-interval measurements, or voltage-to-frequency converters to enable measurements of voltage and resistance.

The 5312A ASCII Interface is addressable, meaning that a controller can direct any instrument using this module to place measurement data on the bus while the others remain silent. In keeping with the goal of providing accuracy and reliability at minimum cost, however, the 5300 instruments are not programmable, so functions and ranges must be set manually (except for those that are autoranging).

The modules are compatible with the various possible levels of system complexity. In the "talk only" mode they can be used with an HP-IB compatible printer (e.g., Model 5150A, page 14), to provide an economical data logger. In the "addressable" mode, up to 13 may be used with the Model 5150A Printer and its optional scanner to collect data from several sources that are making the same or completely different measurements from many inputs simultaneously, such as voltages from strain gauges and thermocouples, frequencies from carrier pre-amps and rotating machinery, or time intervals from photocells. Each instrument can gather data continuously, as it would in a stand-alone application, but it places data on the interface bus only when addressed by the controller. If the timing of a measurement is important, such as several inputs to be measured simultaneously, an initiate command directs each instrument to take a reading and hold it until addressed to put the data on the bus. When used with a desk-top calculator or a computer, the modules provide a low-cost approach to multiple-input data acquisition systems that can be modified any time with more complex instruments without modifying the interface.



Because the output of the ASCII Interface module is byte-serial, it is possible to output more information than is possible with the familiar BCD output. The module's output is 15 bytes in a format like the following, as it would appear on a Model 5150A Thermal Printer (it also transmits the LF and CR commands):



Inclusion of an indication of measurement units is especially useful with multifunction modules, such as the Model 5306A Multimeter (volts, ohms, frequency).

# Visualizing Interface Bus Activity

*Connecting to the HP Interface Bus, a new analyzer listens to and displays the status of all bus lines for easy study of bus activity. It also serves as a talker, using programs in its internal memory to exercise bus-compatible instruments and systems.*

**by Harold E. Dietrich**

AS ANYONE WHO HAS ASSEMBLED instruments into a system well knows, software and hardware problems always seem to arise. Many of these problems are avoided when the HP Interface Bus is used but, even though the bus standardizes connectors, control lines, signal levels, and message transfer protocol, software errors can occur if the system designer does not completely understand the bus system or the capabilities of the instruments he's using. Hardware problems occur if the instruments are not functioning properly or if they are not completely compatible with the bus standard. Additional problems face the designer of a new instrument when he evaluates its compatibility to the interface bus system.

Solutions to these problems are found much more quickly with the help of the new Model 59401A Bus System Analyzer (Fig. 1). It connects to the bus in the same way as any other instrument. As a bus listener, it displays the status of all the lines in the bus. It enables the designer to go through his program step by step and, by making bus traffic observable, it makes software debugging relatively easy.

Besides serving as a listener, the analyzer can also

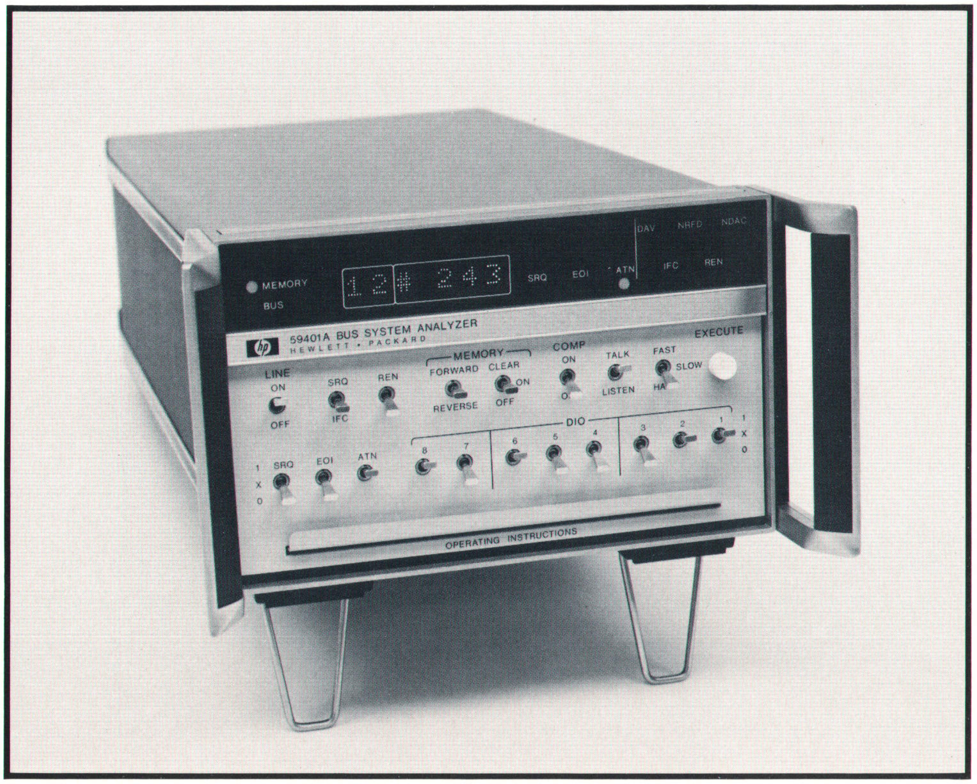


**Fig. 1.** *Model 59401A Bus System Analyzer monitors activity on the HP Interface Bus, displaying the instantaneous status of all bus signal lines. It can also be a bus controller using programs stored in its memory or program steps set up on the front-panel switch register.*

be a talker. It thus can completely exercise another talker, listener, or controller. It has an internal read-write memory that holds 32 program steps, loaded either from the front-panel switch register or from the bus itself. With a suitable program loaded, the Analyzer can exercise instruments at maximum bus speed, or step by step.

### Visibility

Bus activity is made visible by an array of displays on the analyzer's front panel. Annunciator LEDs indicate the presence of the "true" state on the corresponding bus control lines. Information on the eight data lines, however, is converted to the equivalent octal number for numeric display, making it easier to read the instantaneous contents of the bus. As a further aid, the equivalent ASCII character is also displayed, which makes it easier to follow the steps in a calculator-based program.

The same front-panel alphanumeric display is also used for display of the contents of the internal memory. Two additional digits are then illuminated to show the address in memory of the character on display.

### Monitoring Traffic

In the LISTEN mode, the bus system analyzer functions as an addressed listener to all bus traffic. Because data transfer on the bus is under control of the three "handshake" lines (see box, page 3 ), the analyzer can readily slow traffic to a speed convenient for visual monitoring. Three "speeds" are provided. In the FAST mode, bus traffic proceeds at the fastest rate allowed by the slowest instrument in the system. In the SLOW mode, the bus analyzer limits bus speed to two characters per second, a rate that enables visual monitoring of bus traffic. In the HALT mode, a character is accepted only when the analyzer's EXECUTE button is pressed, allowing the program to be stepped one character at a time. The operator can switch from SLOW to HALT when bus traffic approaches the place in the program where a software problem is known to exist, and then proceed one step at a time to find the problem. It is this ability to go through a program step by step while displaying bus activity that gives the analyzer its great usefulness.

The bus analyzer also has a COMPARE mode. It stops bus traffic when the character on the bus matches the settings of the front-panel switch register (the lower row of switches, Fig. 1). A program thus stopped may be resumed by pressing the EXECUTE button.

Whether or not the COMPARE switch is on, the analyzer outputs a pulse whenever a bus character matches the front-panel switch register. This can be used to trigger an oscilloscope to observe related analog signals or to examine transients on the bus (test points for all bus lines are available at the rear of the instrument).

### Replay of Past Events

The bus analyzer can store data characters as they appear on the bus, continuously updating its memory so it always contains the most recent 32 characters. The COMPARE mode can be used to stop bus traffic where a problem is known to exist, then the previous 32 steps, now stored in memory, can be recalled one by one to determine what led to the problem. A front-panel switch steps the memory in either a forward or reverse direction.

For example, a DVM in a system occasionally outputted a negative reading when all readings were supposed to be positive. The analyzer's switch register was set to the ASCII code for "-" ($00\ 101\ 101_2$ or $055_8$) and the system was operated with the analyzer in the COMPARE mode. The occurrence of the minus symbol on the bus data lines halted bus traffic, and the previous steps could then be recalled from the Analyzer's memory to find out what happened.

### Step by Step

The TALK mode is useful in checking out a new interface design. The analyzer can determine whether or not an instrument can be addressed and unaddressed, whether or not it meets timing requirements (sufficiently short settling time, releases lines in time, etc.), and whether or not it can handshake with other instruments that are either faster or slower.

For example, a calculator-based system using several programmable multiposition switches was assembled. At one part of the program, the software designer wished to program two switches to the same position, so the program addressed both to LISTEN followed by programming information. The following calculator command statement was used:

CMD "?U#$", "A2B2"

The symbol "?" is the unlisten command and "U" designates the calculator as the talker. "#" is the listen address of the first switch, "$" is the listen address of the second, and A2B2 programs both for switch position 2. But, when the program was executed, only the second switch programmed correctly.

To trace the problem, the first switch was connected to the bus analyzer. The analyzer was set to TALK and the switch's listen address (# or X0 100 011) was set into the analyzer's front-panel switch register. With the analyzer's ATN switch set to true, DAV was driven true when the EXECUTE button was pressed, thus sending the address. The switch responded by driving NRFD true and NDAC false, as shown by the analyzer's front-panel annunciators.

When the EXECUTE button was released, DAV went

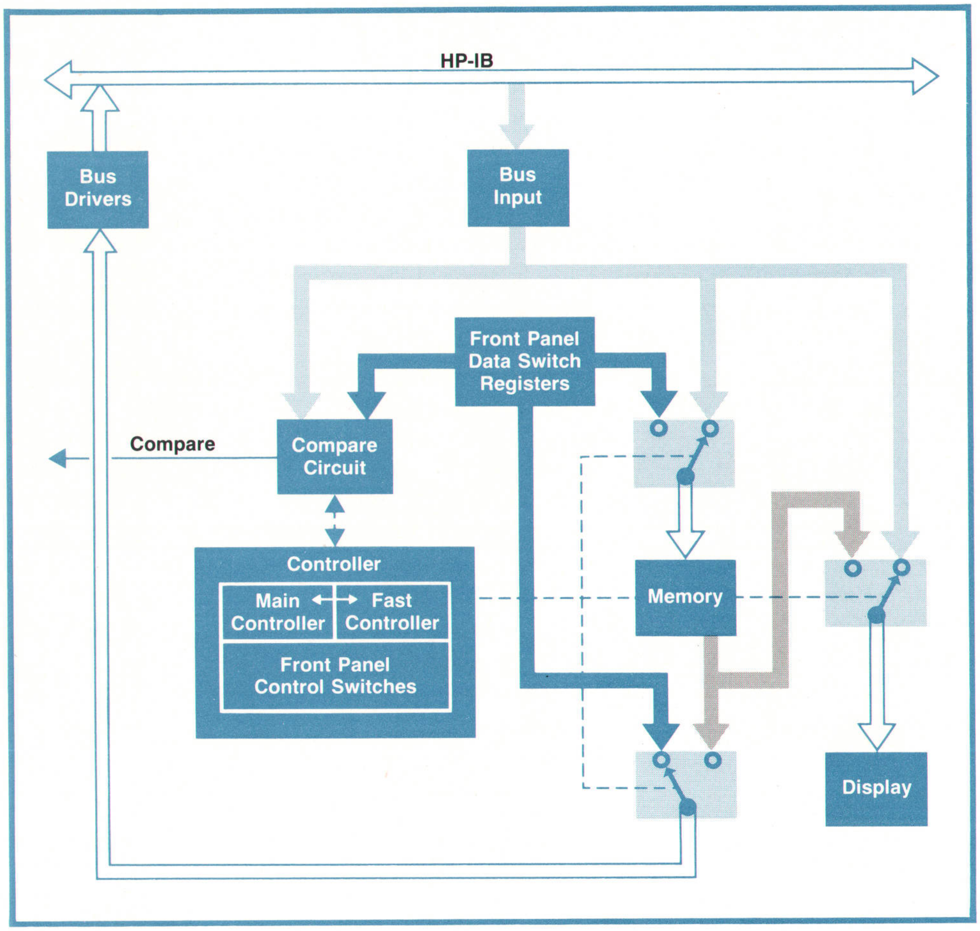


**Fig. 3.** *Simplified block diagram of the Model 59401A Bus System Analyzer. The switches shown represent gates operated in combinations by the internal controller in response to front-panel switch settings. The main controller manages most operations but a fast bipolar 'controller is used for the FAST TALK and FAST LISTEN functions.*

false. The switch then sent NDAC true and NRFD false, indicating that it had accepted the address and was ready for more inputs. ATN was then set false, and NDAC remained true, indicating that the switch was addressed to listen.

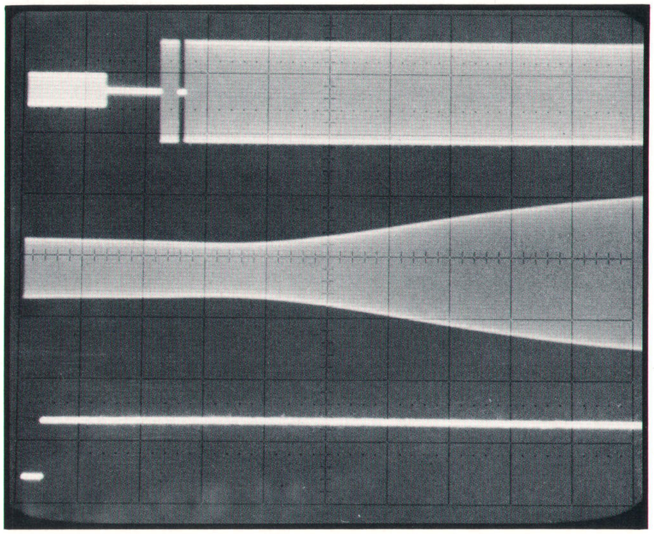


**Fig. 2.** *Oscillogram demonstrates use of the bus analyzer as an oscilloscope trigger (see text). The bottom trace shows the timing of the analyzer's "compare" pulse. The top trace shows the synthesizer output in response to the trigger event and the middle trace shows the output of the filter under test.*

The programmer then set the address of the second switch on the analyzer's front-panel register and repeated the procedure. This time, when ATN was set false, NDAC went false. This showed that the first switch became unaddressed when another listen address was placed on the bus. The problem was thus traced to the interface logic in the first switch.

### Speed-Related Problems

A rear-panel input for an external clock enables the analyzer to slow the rate of bus activity to any rate below system maximum. This was used to track down a problem with a listener that would operate with some controllers but would cause the bus to hang up when used with others.

The analyzer was connected to the listener and the controller program was loaded into the analyzer's memory. This "minisystem" was then operated at various data rates by varying the external clock rate until the fault occurred. The problem was found in the handshake response and was quickly traced to an optical-isolator circuit.

### Ancillary Activities

The photo in Fig. 2 illustrates how the Analyzer can be used for observing analog phenomena related to interface bus activity. The need was to determine what effect, if any, programming data for a frequency synthesizer had upon a test of a crystal filter. The test was to determine the response of the filter to a 10-dB change in signal amplitude. Filter center frequency was 100 kHz and bandwidth was 100 Hz.

Programming information for the synthesizer was stored in the analyzer's memory. To obtain a compare pulse for triggering the oscilloscope, a "1" was placed on data line DIO8 with the data byte that executes the change to the new output amplitude. The synthesizer ignores line DIO8 since it responds only to 7-line ASCII characters. The analyzer's switch register was set to 1XXXXXXX.

As the program executed, the oscilloscope triggered on the step that included the "1" on line DIO8. As the oscillogram of Fig. 2 shows, transients are caused by the switching of relays in the synthesizer's programmable step attenuator when a 10-dB or greater change in amplitude is called for.

## Internal Organization

A simplified block diagram of the Model 59401A Bus System Analyzer is shown in Fig. 3. As a listener, the instrument accepts inputs from all the bus lines and displays their status on the annunciator LEDs and alphanumeric display. As a talker, characters and control signals are placed on the bus either from the analyzer's memory or from the front-panel switch register.

The memory consists of six 64-bit RAMs. In the listen mode, the memory is operated in push-down fashion so the oldest data is dropped as new data comes in. In the talk mode, characters set on the front-panel switches are entered in by the EXECUTE button. The contents of any memory location can be edited by using the FORWARD/REVERSE switch to step the memory to the address desired, then pressing the EXECUTE button to enter the new switch-register data.

As the block diagram shows, two controllers are used. The "fast" controller enables the analyzer to operate up to the maximum speed the bus may ever operate (1 megabyte/s). To accommodate the handshake operation, this speed requires the use of a bipolar controller. However, performing all operations within the instrument with a bipolar controller would have been excessively expensive. Hence, the bipolar controller is used only for the FAST LISTEN and TALK modes and an MOS controller is used for all the other functions.

To minimize the number of state times needed for processing a bus character, the fast controller can check up to four inputs (or qualifiers) in each state. For example, one of the states allows the controller to

## ABRIDGED SPECIFICATIONS

### Model 59301A ASCII/Parallel Converter

**INPUT:** From HP Interface Bus; 24-pin connector.

**OUTPUT:** Logic 0 = 0.4V; logic 1 = 5V. Two 50-pin connectors.

PRINT COMMAND: Negative pulse, +5V to +0.4V for minimum of 20 $\mu$s.

INHIBIT COMMAND INPUT: Voltage level within +2.4V to +20V into 5k ohms.

**DIMENSIONS:** ½-width module; 3 lb 12 oz (1.70 kg).

**PRICE IN U.S.A.:** $550.

### Model 59303A Digital-to-Analog Converter

**INPUT:** From HP Interface Bus; 24-pin connector.

**OUTPUT:** −9.99V to +9.99V; front-panel banana plugs in parallel with rear-panel BNC.

**ACCURACY:** Output within ±0.1% (±½ LSD) over 0 to 50°C temperature range.

**SETTLING TIME:** <30 $\mu$s to ±½ LSB.

**BLANKING:** TTL signal that indicates digital information is changing. Duration, 25 $\mu$s after LF is accepted.

**DIMENSIONS:** ¼-width module; 5 lb 12 oz (2.61 kg).

**PRICE IN U.S.A.:** $850.

### Model 59304A Numeric Display

**INPUT:** From HP Interface Bus; 24-pin connector.

**DISPLAY:** Gas discharge (orange) 0.4 in high; 12 characters and decimal point.

**DIMENSIONS:** ¼-width module; 2 lbs 11.5 oz (1.23 kg).

**PRICE IN U.S.A.:** $650.

### Model 59306A Relay Actuator

**OUTPUT TERMINALS:** Banana jacks arranged on the rear panel in three rows, A, B, and C (common). Normally closed position is B to C and normally open position is A to C.

**RELAY CONTACTS:** 0.5A at 28V dc or 115V ac.

**RELAY SETTLING TIME:** 20 $\mu$s.

**DIMENSIONS:** ½-width module; 5 lb 13 oz (2.64 kg).

**PRICE IN U.S.A.:** $650.

### Model 59307A VHF Switch

**IN/OUT TERMINALS:** BNC's, one of four for each input (or vice-versa) selectable under front-panel or program control.

**RELAY CONTACTS:** 0.5A at 25V, <1 ns transition time.

**RELAY SETTLING TIME:** 20 $\mu$s.

**VSWR:** <1.1.

**ISOLATION:** >40 dB @ 100 MHz.

**DIMENSIONS:** ½-width module; 7 lb 2 oz (3.23 kg).

**PRICE IN U.S.A.:** $750.

### Model 59308A Timing Generator

**TIME INTERVAL ($\Delta$t) RANGE:** 1 $\mu$s to 99,900s (001E0 to 999E8 $\mu$s), selected on thumbwheel switches or programmed through HP Interface Bus.

**TIMER MODE:** Outputs one pulse one $\Delta$t following start trigger.

**TRIGGER OUTPUT (rear panel):** TTL or ECL logic levels (switch-selected), 50 ns transition time. Pulse width, 500 ns ± 100 ns (pulse mode) or ½$\Delta$t (square-wave mode).

**TRIGGER INPUT (rear panel):** Edge triggered at 0.5V or 2V, positive or negative slope, switch-selected. Input R, 10 k$\Omega$.

**TRIGGER/RESET PUSHBUTTON (front panel):** Output trigger occurs within 1 $\mu$s of time set after pushbutton is released.

**TIME BASE:** Crystal frequency, 10 MHz; 3 parts in $10^7$ per month aging rate; temperature, ±5 parts in $10^6$, 0° to 50°C. Rear-panel BNC accepts 1, 5, or 10 MHz external standard.

**DIMENSIONS:** ½-width module; 4 lb 10 oz (2.10 kg).

**PRICE IN U.S.A.:** $875.

### Model 59309A ASCII Digital Clock

**DISPLAY:** Month, day, hour, minute, second. LED numerals. }
**RESET:** Resets display to 01:01:00:00:00 and starts clock. } Programmable

**SET DAY/TIME:** Updates display (fast or slow) to arrive at desired time and date.

**LEAP YEAR:** Switch selects 365 or 366 days/year (non-programmable).

**ERROR INDICATOR:** All decimal points light to indicate possible error from power interruption or missed counts.

**TIME BASE:** 1-MHz room-temperature crystal; 5 parts in $10^6$ aging rate; temperature, 5 parts in $10^6$, 10° to 40°C (0.5 s/day). Accepts external 1, 5, or 10 MHz frequency standard (1V rms into 1 k$\Omega$).

**STANDBY POWER:** Internal 9V battery (not supplied) maintains time for 1 day with display off. Accepts 8 to 10V at 2 mA from external source.

**DIMENSIONS:** ¼-width module; 2 lb 10 oz (1.18 kg).

**PRICE IN U.S.A.:** $975.

### All 59300 Series Units

**DIMENSIONS:**

WIDTH: Quarter-width modules, 4.17 in (105.9 mm). Half-width modules, 8.38 in (212.9 mm).

HEIGHT: 4 in (101.6 mm).

DEPTH: 11.4 in (289.6 mm).

**POWER:** 115/230V ± 10%, 50 to 400 Hz; 15 VA max.

### Model 5312A ASCII Interface

**SAMPLE RATE:** Controlled by mainframe front panel control or by setting rate of reset command (when in listening mode, counter can be reset by sending the letter I).

**TRANSFER TIME:** 20 milliseconds (typical).

**TRANSFER RATE:** Maximum of 40 reading/sec depending on capabilities of plug-on.

**SELF TEST MODE:** Checks functioning of interface.

**NOTE:** The 5312A is not compatible with the 5300A mainframe which contains its own BCD Digital Output.

**PRICE IN U.S.A.:** $350.

### Model 5150A Thermal Printer

**PRINTING TECHNIQUE:** Thermal print, 5 × 7 dot matrix.

**RATE AND SPACING:** 3 lines/s, 6 lines/inch.

**PAPER:** Thermal sensitive in rolls or fan-folded.

**POWER:** 100, 120, 220, or 240V, 48 to 440 Hz (50 or 60 Hz only with clock option); 100 VA.

**DIMENSIONS:** 8.5 in W × 7.5 in H × 14.25 in D (216 × 178 × 356 mm).

**WEIGHT (with one option):** Approximately 16 lbs (7 kg).

#### ASCII Interface (opt 001)

**COLUMNS:** 20.

**CHARACTER SET:** 64 ASCII characters (columns 2, 3, 4, and 5 of ANS 3.4 – 1968 except "↑" in column 5, row 14).

#### BCD Interface (opt 002)

**COLUMNS:** 10 (20 with two option 002's installed).

**CHARACTER SET:** 0 through 9, +, −, V, A, R, and blank. Special character sets available.

**INPUT:** TTL levels, switch selects + or − true logic. Parallel BCD (8421) format.

**PRINT COMMAND:** TTL level, + or −

#### Scanner (opt 003)

**INSTRUMENTS SCANNED:** 1 to 13.

**CYCLE TIME:** Limited by slowest of (1) instrument response time, (2) 3 samples per second, or (3)DATA PRINT INTERVAL with optional clock.

**COMPATIBILITY:** HP Interface Bus.

#### Clock (opt 004)

**DISPLAY:** 6-digit LED display of hours, minutes, seconds; settable via front-panel switches.

**TIME BASE:** Line frequency, 50 or 60 Hz (selectable by internal jumper).

**DATA PRINT INTERVAL:** Minimum, 1, 2, 10, 20 sec; 10, 20 min; 1, 2 hours.

**TIME PRINT INTERVAL:** Same as data; interlock prevents intervals shorter than data interval.

**PRICES IN U.S.A.:** 5150A Printer, $800, Scanner, $175.
ASCII Interface, $175, Clock, $250.
BCD Interface, $110.

**MANUFACTURING DIVISION:** SANTA CLARA DIVISION
5301 Stevens Creek Boulevard
Santa Clara, California 95050

## SPECIFICATIONS

### HP Model 59401A Bus Systems Analyzer

**LISTEN:**

ACCEPT TIME: <750 ns

READY TIME: <750 ns

**TALK:**

1. Data changed >500 ns before DAV pulled low.
2. ATN driven low >1 $\mu$s before DAV pulled low.
3. DAV driven high <700 ns after NDAC is false.
4. DAV driven low <700 ns after NRFD is false if conditions 1 and 2 are met.

**EXTERNAL CLOCK INPUT:** TTL gate input; ≤10 MHz repetition rate.

**COMPARE OUTPUT:** TTL gate output, low true.

**POWER:** 100/120/220/240V, +5%, −10%, 48 Hz to 66 Hz; 42 VA max.

**DIMENSIONS:** 5.73 in H × 9.65 in W × 19.5 in D (145 × 245 × 495 mm).

**WEIGHT:** 12 lb 7 oz (5.64 kg).

**PRICE IN U.S.A.:** $4900.

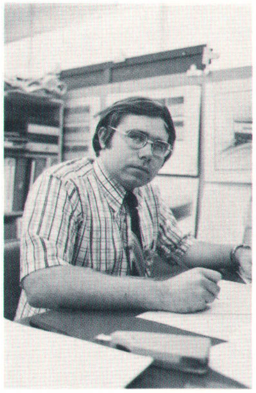
**MANUFACTURING DIVISION:** LOVELAND INSTRUMENT DIVISION
P.O. Box 301
815 Fourteenth Street, S.W.
Loveland, Colorado 80537

check whether listeners are ready for data (NRFD), whether sufficient time has elapsed since the data was changed and since ATN was pulled low, and whether there has been an interrupt from the main controller. If all conditions are met, data valid (DAV) can be driven low within one state time (200 ns). Thus, the fast controller needs only 16 states, and it operates with 1280 bits of ROM. The main controller has 256 states and uses 14,336 bits of ROM.

**Donald C. Loughry**

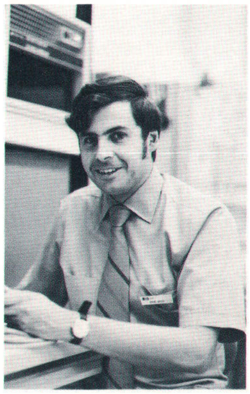
A 1952 graduate of Union College (BSEE), Don Loughry joined Hewlett-Packard as a production test manager in 1956. Two years later he moved to the Dymec Division lab to work on systems and later became division engineering manager. In 1968, Don was appointed corporate interface engineer. He's active in various groups working on interface standards, including IEC, ISO, ANSI, and IEEE. Don is also interested in skiing, photography, gardening, and church work.

**Steven E. Schultz**



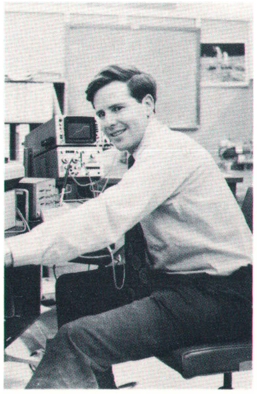
Steve Schultz joined HP in 1971, going to work for the 5345A Counter group before becoming project leader on some of the ASCII-Programmable Modules. A 1970 graduate of the University of California at Berkeley (BSEE), Steve is now finishing work on an MSEE at Stanford in the HP Honors Co-op program. Most of his spare time goes into his studies but he has many other interests, including dirt bikes and radio-controlled boats. He and his wife live in Menlo Park, California.

**David W. Ricci**



One of the developers of the HP Interface Bus, Dave Ricci sometimes participates in the deliberations of various standards committees. He joined HP in 1965 after earning his MSEE degree at the University of California at Berkeley (he obtained his BSEE at the California State Polytechnical University). Initially he worked on Multichannel Analyzers but he is now developing applications for counters that use the HP-IB. Dave and his wife like to ski, sail, and backpack. They have one son, 2½ years old.

**Charles R. Trimble**

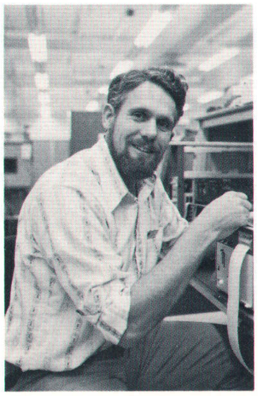


Graduating with honors from the California Institute of Technology in 1963, Charlie Trimble went on to get his MSEE there the following year. He then joined Hewlett-Packard where his initial project was the 5480A Signal Averager. Now an engineering section manager, he has a dual responsibility for precision timing measurements and for the Santa Clara Division's HP-IB activities. Charlie and his wife like sailing, backpacking, and bicycling. They live in Los Altos, California.

**Peter S. Stone**

Pete Stone did it all in one stretch: BSEE, MSEE, and PhD degrees at the University of Minnesota. He then worked for a medical equipment firm for a year before joining Hewlett-Packard in 1972 to work on automatic systems. He is presently group leader in the voltmeter lab. Married, but with no children yet, he still finds time for flying (he's 1/3-owner of a Cherokee 180) and ham radio (both fixed and mobile).

**Hans-Jürg Nadig**



Hans Nadig came to the United States and Hewlett-Packard in 1967. Initially he worked on Fourier Analyzers, then spent a year in IC production before taking on the thermal printer project. Hans graduated with a Diplom Ingenieur from the Federal Institute of Technology in Zurich, Switzerland, and designed automatic controls for mass transit systems before joining HP. He has a wife and two daughters and likes ski touring, mountain climbing, and photography.

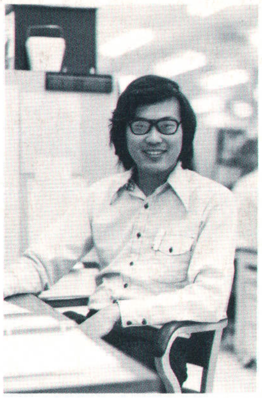
### Acknowledgments

Thanks are due Michael C. Williams for the product design and Skip Beatty for help with the breadboard. Gene Meismer and Al Boswell aided the transition into production. Special thanks are due Gerald E. Nelson for his guidance.

**Harold E. Dietrich**

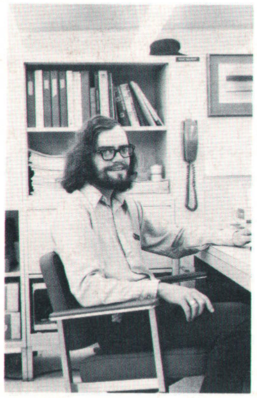
A 1966 graduate of Purdue University (BSEE), Harry Dietrich came straight to work for Hewlett-Packard, contributing to the 3555A Telephone Test Meter, 653A Video Test Oscillator, and 3570A Network Analyzer before taking on the bus analyzer. Harry earned his MSEE degree at Colorado State University in 1970 in the HP Honors Co-op program. He likes hunting but he also takes his family, which includes three small sons, on camping trips in his pickup camper.

**Gary D. Sasaki**



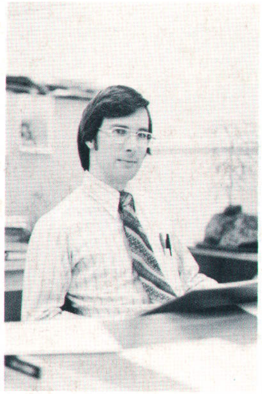
Gary Sasaki graduated from the University of California (Berkeley) with a BSEE in 1973 and came straight to work on the ASCII Interface for the 5300 Measuring System. Gary actively promotes the engineering profession to high school students by giving talks at schools and holding informal "rap" sessions in the plant with students. For fun he plays volleyball and ping-pong. He and his wife build furniture by hand (their only power tool is a drill).

**David L. Wolpert**



Dave Wolpert joined the Hewlett-Packard Loveland Instrument Division in 1972 upon getting a BSEE degree at the Georgia Institute of Technology. Initially he worked on some investigatory projects, then went to the 3459A Scanner project. Dave's married and he gets involved in photography, doing his own darkroom work, and folk guitar.

**Lawrence P. Johnson**



Larry Johnson earned a BSME at Rensselaer Polytechnic Institute followed by a Master's degree in manufacturing engineering at Boston University (1968) and later an MBA. He joined HP's Medical Electronics Division in 1968 as a process engineer, spent some time at the San Diego Division, then moved to the Santa Clara Division as product manager for the 5300 Measurement System, which included work on the ASCII Interface. Larry and wife ski and dabble in real estate.

Hewlett-Packard Company, 1501 Page Mill
Road, Palo Alto, California 94304

HEWLETT-PACKARD JOURNAL
**JANUARY 1975** Volume 26 • Number 5

**Technical Information from the Laboratories of
Hewlett-Packard Company**

Hewlett-Packard S.A., CH-1217 Meyrin 2
Geneva, Switzerland
Yokogawa-Hewlett-Packard Ltd., Shibuya-Ku
Tokyo 151 Japan

**Editorial Director • Howard L. Roberts
Managing Editor • Richard P. Dolan
Art Director, Photographer • Arvid A. Danielson
Illustrator • Sue M. Perez
Administrative Services, Typography • Anne S. LoPresti
European Production Manager • Michel Foglia**

CHANGE OF ADDRESS: To change your address or delete your name from our mailing list please send us your old address lable (it peels off). Send changes to Hewlett-Packard Journal, 1501 Page Mill Road, Palo Alto, California 94304 U.S.A. Allow 60 days.