HEWLETT-PACKARD

JOURNAL

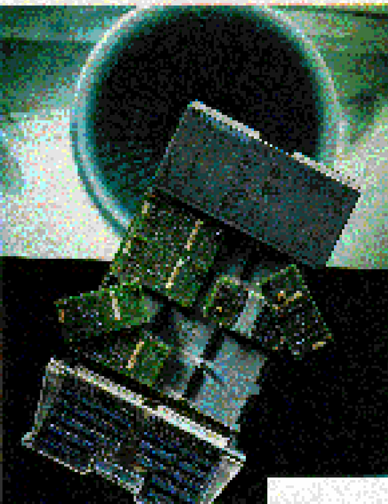


table of contents

October 1994,
Volume 45, Issue 5

Articles

1. **Customer-Driven Development of a New High-Performance Data Acquisition System,** *by Von C. Campbell*

2. **A Compact and Flexible Signal Conditioning System for Data Acquisition,** *by John M. da Cunha*

3. **High-Throughput Amplifier and Analog-to-Digital Converter,** *by Ronald J. Riedel*

4. **On-the-Fly Engineering Units Conversion,** *by Christopher P.J. Kelly*

5. **Built-In Self-Test and Calibration for a Scanning Analog-to-Digital Converter,** *by Gerald I. Raak and Christopher P.J.*

6. **Manufacturing Test Optimization for VXI-Based Scanning Analog-to-Digital Converters,** *by Bertram S. Kolts and Rodney K. Village*

7. **Design Leverage and Partnering in the Design of a Pressure Scanning Analog-to-Digital Converter** *by Richard E. Warren and Conrad R. Proft*

8. **Integrated Pin Electronics for Automatic Test Equipment,** *by James W. Grace, David M. DiPietro, Akito Kishida, and Kenji Kinsho*

9. **CMOS Programmable Delay Vernier** *by Masaharu Goto, James O. Barnes, and Ronnie E. Owens*

10 **Real-Time Digital Signal Processing in a Mixed-Signal LSI Test System,**
by Keita Gunji

11 **Vector Error Testing by Automatic Test Equipment,**
by Koji Karube

12 **High-Frequency Impedance Analyzer**
by Takanori Yonekura

13 **Virtual Remote: The Centralized Expert,**
by Hamish Butler

14 **Frame Relay Conformance Testing,**
by Martin Dubuc

15 **The FDDI Ring Manager for the HP Network Advisor Protocol Analyzer,**
by Sunil Bhat, Robert H. Kroboth, and Anne L. Driesbach

16 **FDDI Topology Mapping**
by Sunil Bhat

17 **Automation of Electrical Overstress Characterization for Semiconductor Devices,**
by Carlos H. Diaz

Customer-Driven Development of a New High-Performance Data Acquisition System

The HP HD2000 data acquisition system provides C-size VXIbus modules that are tailored to provide fast and accurate acquisition of temperature, pressure, strain, volts, and resistance data for turbine and piston engine testing applications.

by Von C. Campbell

Products from the past like the HP 2250 measurement and control processor and current products like the HP 3852 measurement/control instruments have given HP a reputation for excellent measurement integrity and throughput. However, within the turbine and piston engine test markets we found certain groups who stated that our measurement products "almost, but not quite" did their job. Obviously there was something in these applications that these products could not do. What was missing? Only customers could tell us.

We undertook an effort to find and talk to turbine and piston engine test customers so that we could understand their applications and the problems they were facing with existing products. These discussions were done in the language of the customer, which required some vocabulary development on our part. After understanding these first applications, we widened our scope to include as many customers as possible. Over 100 customer visits were conducted, including virtually every turbine engine manufacturer and larger piston engine manufacturers worldwide, many being visited more than once.

A very clear picture of these applications emerged. In turbine testing for example, we discovered three use models with similar but distinct measurement needs. The first model was found in engine design and development. In this model large amounts of data (up to thousands of test points per engine) are collected for analysis of the thermodynamic performance of the engine and for verification of the simulation models used in development. The second use model was found in production test of a finished engine immediately after manufacture. Here a smaller test system is used to verify proper engine operation. Finally, the last use model was found in the overhaul and repair of an engine. In this model an even smaller set of data is collected to determine if the engine is performing well enough to be returned to service.

Some common threads ran through all the applications. Accuracy was one of these common features. Temperature measurements to better than 0.25 °C and pressure measurements to 0.05% are required to verify the efficiency of an engine. Engine efficiency is especially important to manufacturers who guarantee fuel consumption rates to their airline customers. Another important factor is the requirement that the test system be capable of being scaled to accommodate from about 100 to over 2000 measurement points for different sized test systems. Continuous high-speed acquisition, in which data is recorded for long periods without interruption, is also an important factor. Measurement rates of up to 1000 Hz per channel on different sensor types allow a better understanding of the static and transient behavior of the engine, but the aggregate throughput of such large, fast systems is a significant challenge. Older measurement systems used many independent instruments to record the volume and variety of data taken during a test. After the test, all the various data records had to be combined to form a single integrated picture of test results. To overcome this problem of data integration it is imperative that all data be measured and recorded deterministically.

One common need that goes beyond the acquisition system is the industry-wide pressure on test departments to be more productive. The desire to reduce test times and the resources needed to install, develop, and operate a test system is universal. In some areas, there was a strong desire to have a third party develop and install the test system so that the organization could concentrate on testing. This information led to our getting in touch with the leading systems integrators to understand how we could help them solve their customers' problems.

After developing a clear picture of user needs and reviewing past implementations, we came to the conclusion that although individual measurement requirements were being met with existing products, system requirements like those mentioned above were not being met. From this conclusion, the HP HD2000 VXIbus-based data acquisition system was born (see Fig. 1).

VXIbus technology[1] has the right capabilities to meet the system issues we encountered during our customer analysis. The VXIbus architecture allows us to integrate mixed measurements from multiple sources onto one computer bus, allowing high speed and determinism in data sampling. Since VXIbus is an open standard we can include non-HP products that offer specific functions like IRIG B time stamping, MIL STD 1553, and ARINC 429 communications instruments into our system. These products help meet overall systems

Fig. 1. The HP HD2000 data acquisition system showing the HP E1413 and HP E1414 modules.

requirements without requiring large additional development time and cost.

The Measurement Modules

The bulk of the HP HD2000 system measurements come in the form of analog input, with measurements of temperature and pressure making up about 90% of the volume of data. To meet these needs the HP E1413 64-channel scanning analog-to-digital converter and the HP 1414 pressure scanning analog-to-digital converter were developed. These converters are designed to maximize measurement accuracy, throughput, determinism, flexibility, and density while minimizing computer resource use, program development time, and cost. These modules are the first two modules of the HP HD2000 data acquisition system.

The single-slot HP E1413 combines a highly accurate 100-kHz, 16-bit, autoranging analog-to-digital converter with 64 channels of a high-speed multiplexer and eight banks of front-end signal conditioning plug-ons (such as amplifiers and low-pass filters) and has an internal calibration source for end-to-end calibration. Gain and offset errors for every channel, including signal conditioning, can be removed through the automatic calibration process, maximizing measurement accuracy. An onboard digital signal processor (DSP) controls the card's basic operation, including sequencing of multiple channel scan lists, real-time limit checking, conversion of data to engineering units, and conversion into a computer-ready, 32-bit floating-point number. These numbers are output into a 65,000-reading FIFO buffer and a current value table, which allows instant access to the most recent reading on any channel. These features minimize the amount of interaction the card needs to have with the host computer and the amount of work it has to do to make the data interpretable. This tightly coupled architecture maximizes throughput while minimizing the need for external computer resources. Fig. 2 shows a block diagram of the HP E1413.

In pressure measurement systems, which typically have a large number of test points, we found that electronic pressure scanning technology is the measurement technology of choice. The recognized leader in this technology is Pressure Systems Inc. (PSI) in Hampton, Virginia. As a continuation of our focus on user needs, we developed a partnership with PSI to develop a pressure scanning analog-to-digital converter. This product, the HP E1414, has all the features of the HP E1413 but is designed to interconnect and operate with PSI's electronic pressure sensors and pressure calibrators. The HP E1414 brings the same level of measurement performance and throughput to pressure measurements as the HP E1413 does for the other analog measurements. It also integrates the deterministic measurements of both temperature and pressure into one common system on the VXIbus backplane.

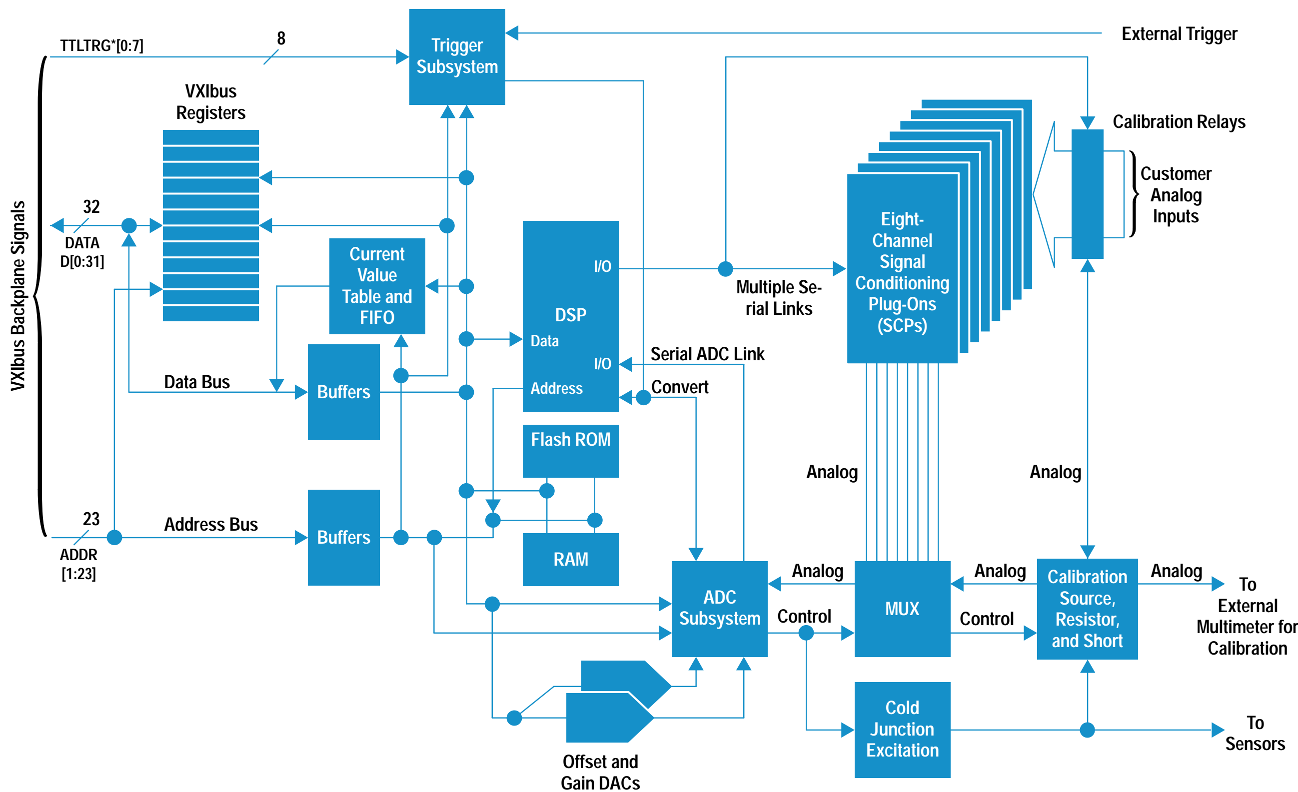


Fig. 2. A block diagram of the HP E1413 64-channel scanning analog-to-digital converter.

The HP E1413 is described in detail in the articles on pages 9, 16, 21, 25, and 30. The HP E1414 is covered in the article on page 35.

To help minimize customers' development time for the acquisition software, all the products in the HP HD2000 family have drivers that maximize their commonality and performance. These drivers allow the instruments to be controlled using the Standard Commands for Programmable Instrumentation (SCPI) language.[2] This open standard language has a programming syntax that is easy to read and understand, and it has a high level of commonality between many different instruments. This minimizes programming time and enhances supportability. To meet our users' needs for high throughput, a C language preprocessor was developed to process the standard SCPI commands into a format that, along with the driver code, can be compiled into high-speed, run-time code. This compiled SCPI (C-SCPI) gives the programmer the ease of programming in a high-level language and the execution speed of assembly code.

Conclusion

The HP HD2000 system began with understanding the users' needs from the perspective of their whole system. This understanding led us to choose the VXIbus architecture, which provides the high throughput, tight coupling, and mixed-measurement capabilities our customers need. Understanding customer needs focused our development efforts on products like the HP E1413 and HP E1414 and guided our software implementation to maximize performance without increasing development time. Finally, a clear idea of customer needs enabled us to avoid making enhancements to the products that were considered irrelevant to our customers. These enhancements frequently add cost and time to development and complexity and confusion to the end user. We let the customer tell us what was needed.

Acknowledgments

I'd like to thank Joe Marriott for all his support during development, Don Miller for his market insight, all the members of the HP E1413 and HP 1414 teams including the C-SCPI developers, and all the people in the Loveland manufacturing center who produce and ship these products every day.

References

1. *Hewlett-Packard Journal,* Vol. 43, no. 2, April 1992, pp. 6-80.
2. S.J. Narciso and G.A. Hill, "The VXIbus From an Instrument Designer's Perspective," *Hewlett-Packard Journal,* Vol. 43, no. 2, April 1992, pp. 15-17.

A Compact and Flexible Signal Conditioning System for Data Acquisition

Because turbine test setups can have up to 1000 test points, special demands are placed on a data acquisition system that must fit a large number of measurement channels into a C-size VXIbus module.

by John M. da Cunha

The HP Model HD2000 data acquisition system is targeted for customers in the turbine test market. This market requires a variety of signal conditioning capabilities to make accurate measurements during turbine tests. Special needs include low cost per measurement point, high density, flexibility, and high performance. To take advantage of the HP HD2000 system's high common-mode rejection analog-to-digital converter (ADC), analog signal conditioning has to be fully differential. The variety of functions needed and constraints on cost, density, and the need for high performance presented quite a design challenge.

Customer Needs in Turbine Test

Turbine test customers use data acquisition systems to characterize, refine, and verify designs of new jet engines and to ensure that engines are performing to specification after overhaul or repair. A typical turbine test system consists of from 100 to 1000 measurement points taking data about engine temperatures, fuel consumption, thrust, pressures, and other items. Data gathered is displayed on control consoles where technicians running the tests can monitor the progress of the test. Data is also sent to disk for storage and later analysis. Because there are so many points in a typical test system, the cost of installing and maintaining each point is high. Thus, low cost per point is very desirable.

Because of the high measurement point count, customers desire that as many channels as possible be put in a single C-size VXIbus module. Therefore, we had to create a design that fits 64 channels of signal conditioning into a single C-size module along with the ADC and the VXIbus backplane interface. The HP HD2000's ADC also has full differential inputs with greater than 110-dB common-mode rejection. The density, cost, and full differential inputs require that signal conditioning circuits be small, inexpensive, and have very high common-mode rejection.

The form factor chosen for these signal conditioning circuits uses a plug-on module for each group of eight channels. Since the HP HD2000 has 64 channels total, this means that each single-width C-size module contains up to eight plug-on signal conditioning circuits with eight channels each (see Fig. 1). These boards are called signal conditioning plug-ons, or SCPs. Fitting eight SCPs into a single C-size slot meant that we had to create a design in which each SCP could only be 2.00 in by 4.075 in with 1.5 square inches of usable circuit area for each channel. Even using surface mount technology, these constraints proved very challenging for the higher-functionality SCPs.

Required Functions

The essential signal conditioning functions required by turbine test customers include the ability to:

- Pass a signal straight to the analog-to-digital converter
- Provide low-frequency, low-pass filtering
- Provide programmable gain and filtering for lower-noise measurements on thermocouples
- Provide temperature measurements with thermistors and resistive temperature devices
- Measure strain gauges.

The signal conditioning plug-ons (SCPs) in the HP Model HD2000 provide these basic functions.

Straight-Through SCP. The simplest function required of an SCP is to pass the input directly through to the multiplexer and analog-to-digital converter. We call this the straight-through SCP. The straight-through SCP has additional functionality. Customers want a way to detect an open transducer connection as well as an overvoltage condition. These functions are provided with simple yet effective circuits that yield very good overall results.

Open transducer detection is provided by a very large-value resistor (100 MΩ) which can be switched to the positive and negative supplies (see Fig. 2). The resistor provides a very small current that charges the input capacitance to the point where an overvoltage error or nonsensical reading will occur, thereby notifying the operator that there is something amiss with the transducer. Normally, open transducer detection is only switched on during calibration or system verification. The small amount of current will cause a slight degradation of measurement accuracy if used during a data gathering run. While it can be turned on during a data run with no other ill effects, such a practice is not recommended.

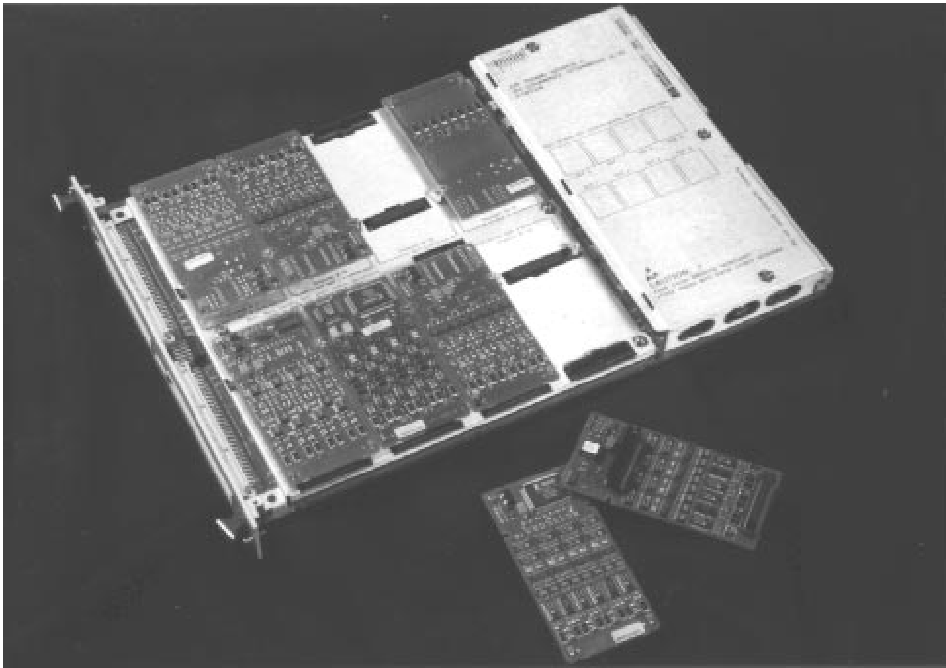


Fig. 1. An HP E1413 module with the cover off showing the eight (two are removed) signal conditioning plug-ons. Each plug-on has eight channels.

Overvoltage protection and ESD (electrostatic discharge) protection are combined into the same circuit for reduced board space and simpler circuitry. Judicious parts placement helped the functionality of the circuits a great deal. The ESD circuit consists of a Zener bias diode with its bias resistor and a large tantalum electrolytic capacitor placed near the ESD ground on the input connector. This provides bias for the diodes that dump static ESD current into the capacitors when the input voltage exceeds ±16.7 volts. ESD current is limited to a manageable value by the physically large, 1-watt, 150-ohm resistors in series with the diodes. Overvoltage protection is provided by monitoring the current through the Zener diode with a transistor. If sufficient current is dumped through the Zener diode, the transistor turns on, sending an interrupt to the digital signal processor (DSP). The DSP will then open protection relays provided on the main board to protect the SCP, the multiplexer, and the ADC. The overvoltage protection feature can also be overridden. Customers override the protection when the cost of stopping the test is more expensive than the measurement system. A fuse resistor is provided to protect the DSP from problems caused by very high overvoltages when the protection feature is overridden. The DSP input is protected by a pair of Schottky diodes that clamp the input. The fuse resistor will open if damage occurs to the input circuits that could put excessive voltage on the DSP input.

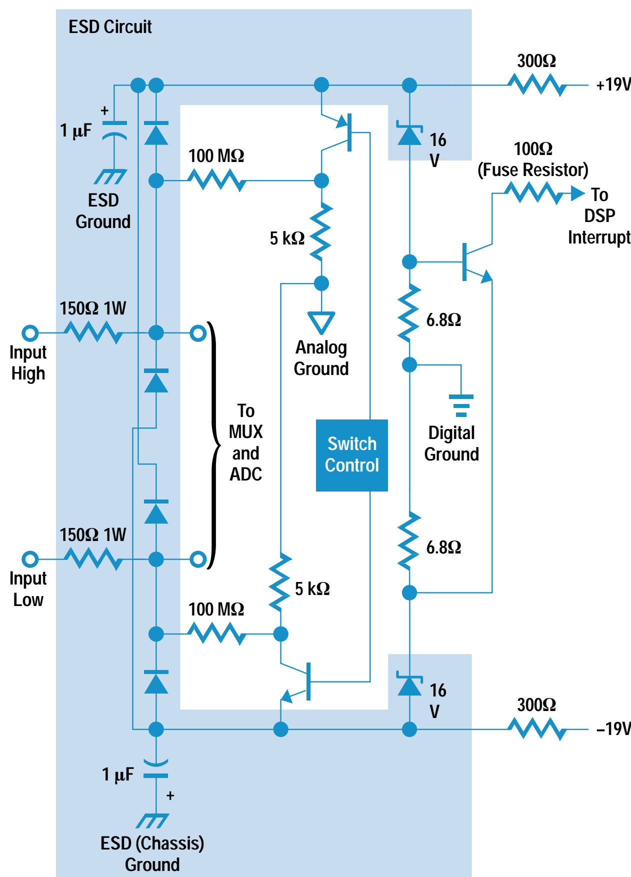


Fig. 2. Open transducer detect, ESD, and overvoltage protection circuit.

Each SCP looks like a memory address segment to the DSP. This memory address segment is called the digital interface address space (see Fig. 3). This address space is divided into two parts on each SCP. One part, called the module space, consists of 64 registers per SCP that are dedicated to addressing functions common to an entire SCP such as the plug-on identifier and scale registers. The other part of the address space, which is called the channel space, consists of 64 registers configured as eight registers per SCP channel.

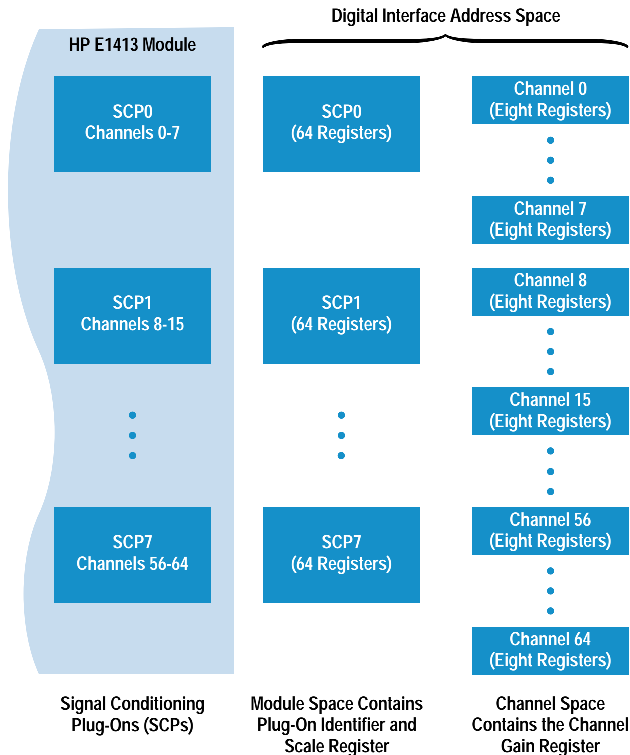


Fig. 3. Digital interface address space.

This address space is used to address functions unique to each of the eight SCP channels such as channel gain and filter setting.

Fixed Filter SCP. A second function required for turbine test is low-frequency, low-pass filtering. Many of the signals from the engine under test represent temperatures measured with thermocouples. Signal voltages are in the millivolt range and subject to high-frequency noise. Low-pass filters with cutoff frequencies of 10 Hz or below are required.

The HP E1413 has only one ADC. Thus, many SCP channels are scanned and multiplexed to the ADC (see Fig. 2 on page 8). With this setup channel-to-channel charge injection through the multiplexer causes errors during fast scanning with passive low-pass filters because the low-pass capacitors hold the charge and cannot dissipate the error charge except through the source and low-pass resistors. For example, if channel 10 on some SCP is holding 6 volts and channel 11 on the same SCP has –15 volts, there is a 21-volt swing when scanning from channel 10 to channel 11, which could cause an error during a fast scan. These errors make it necessary to reduce the scanning speed to achieve the required accuracy. The solution is to buffer the low-pass filter components with an amplifier to absorb the charge injection (see Fig. 4).

As previously stated, very high common-mode rejection must also be preserved in the differential signal path. While it is very desirable to have a sharp cutoff for the filter, there are no practical circuits that can provide a sharp cutoff with an RC type of filter and still produce 100 dB of common-mode rejection. The compromise solution is to use a two-pole

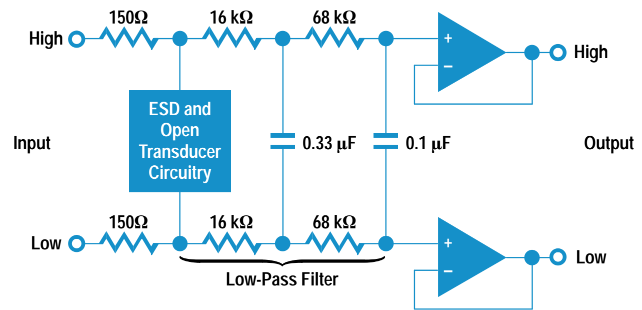
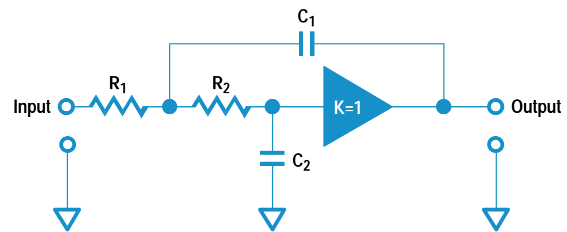


Fig. 4. Fixed filter SCP.

passive RC filter in front of a unity-gain buffer. Care must be taken in choosing pole positions to provide low enough resistances so as not to introduce further errors caused by bias current offsets in the buffer amplifiers. Overvoltage protection and open transducer detection must also be provided. The total solution is a simple, but functional circuit that is very easily changed for different cutoff frequencies.

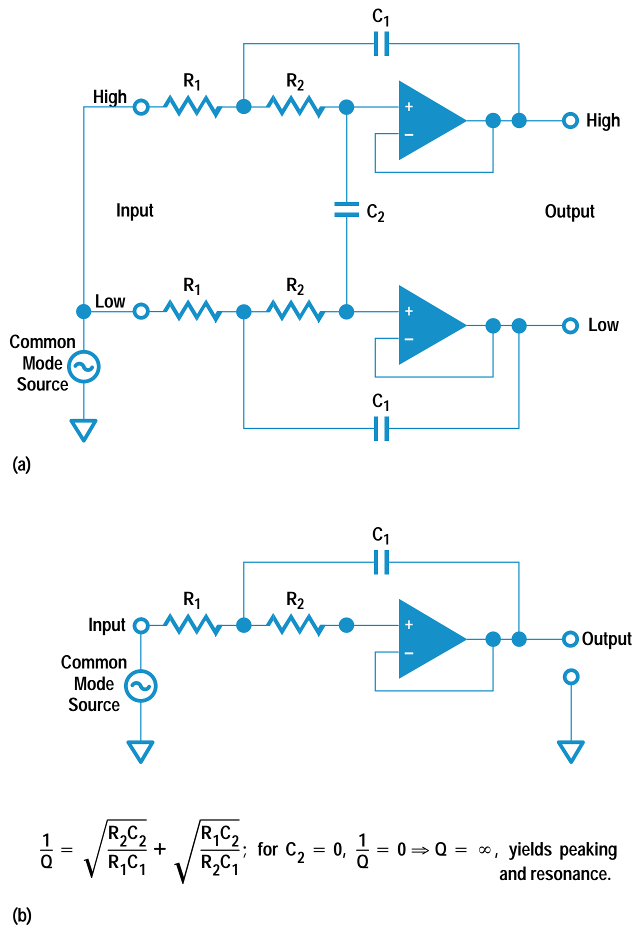
We encountered an interesting phenomenon while developing filter circuits for the fixed filter SCP. A reasonable approach to designing a differential filter is to begin with a single-ended filter (one with the common node at ground) and then transform it into a differential filter. This was the approach originally used to design the fixed filter. The original filter topology chosen was a traditional Sallen and Key structure[1] (see Fig. 5). This filter topology was chosen because of its low parts count and low sensitivities to finite operational amplifier gain-bandwidth and parts variations. The transformation to a differential filter was straightforward and yielded the circuit shown in Fig. 6a. This circuit performed as designed for differential signals but showed anomalies when tested for common-mode rejection.

A spectrum analyzer placed between either output and ground showed a resonance and severe peaking. This anomaly was originally believed to be the result of nonideal behavior in the circuit components. Further investigation showed that this was not the case. There are intrinsic problems with common-mode rejection when using a simple single-ended-to-differential transformation. The equivalent circuit of the



$$T(s) = \frac{K\omega_n^2}{s^2 + \left(\frac{\omega_n}{Q}\right)s + \omega_n^2}, \quad \text{where} \quad \begin{cases} \omega_n^2 = \dfrac{1}{R_1R_2C_1C_2} \\ \dfrac{1}{Q} = \sqrt{\dfrac{R_2C_2}{R_1C_1}} + \sqrt{\dfrac{R_1C_2}{R_2C_1}} \end{cases}$$

Fig. 5. Sallen and Key filter topology.



$$\frac{1}{Q} = \sqrt{\frac{R_2C_2}{R_1C_1}} + \sqrt{\frac{R_1C_2}{R_2C_1}}; \text{ for } C_2 = 0,\ \frac{1}{Q} = 0 \Rightarrow Q = \infty,$$

Fig. 6. (a) The circuit resulting from transforming the filter in Fig. 5 to a differential filter. (b) Equivalent circuit for one half of differential filter.

differential filter with a common-mode signal shows the problem (see Fig. 6b). The behavior of the circuit is such that the voltage drop across C_2 is essentially zero making it appear that C_2 does not exist. In the transfer function C_2 shows up in the damping factor of the complex poles in the half-circuit. As C_2 goes to zero, the damping of the poles goes to zero causing a resonance and the peaking observed. No practical alternative structure was found that did not have a similar difficulty in the transfer function for common-mode signals. As a result of this finding, the circuit in Fig. 4, which has the simpler passive filter with buffer topology, was chosen.

Programmable Gain and Filter SCP. Turbine test customers also want signal conditioning that includes gain as well as filtering to make lower-noise measurements on thermocouples. They want the flexibility to choose different gains and filter cutoff frequencies programmatically. They want this functionality in the same size and channel counts as the straight-through and fixed filter signal conditioning plug-ons. Including this additional functionality and all the other features mentioned above in the same space represented quite a challenge.

The ADC subsystem in the HP E1413 requires gain in powers of two for proper functioning of the engineering unit algorithm, which converts measured voltage data to units such as ohms or degrees (see article on page 21). Signal gains of 8 and 64 were chosen for the amplification factors. Gains of powers of two allow the engineering unit algorithm to adjust the ADC reading to the actual system gain by simply shifting bits up or down. Since a shift operation is faster to execute than a multiply, the engineering unit algorithm can work at the full speed of the ADC. Gains of 8 and 64 also provide sufficient gain to achieve significant noise reductions without overly complex circuitry.

Since programmability is a requirement, gain and frequencies are switched by analog IC switches. The circuit topology takes advantage of the switch density and configuration (dual 4:1 multiplexer). For the lowest-noise performance and most compact design, all the amplification is done in the first stage of the SCP (see Fig. 7). This allows the use of a single analog switch package to control the channel gain. An additional benefit is that the following low-pass filter section reduces the noise bandwidth of the system after the gain stage, thereby reducing the system noise. This topology is also used in other nonprogrammable gain and filter SCPs.

The filter circuit is the same circuit used in the fixed filter SCP, except that it is programmable. Different resistors are switched into the circuit to change cutoff frequencies. The resistors for the programmable gain and filter circuits are built into custom thin-film resistor packs to conserve space and improve thermal tracking performance. Putting all this circuitry into only 1.5 square inches of printed circuit board space required careful layout and frequent consultation with the production engineer assigned to the project. The resulting SCP stretches the limits of current surface mount technology yet still retains a good measure of producibility.

Current Source SCP. Turbine test customers need to measure temperatures with thermistors and resistive temperature devices, and resistances of certain types of sensors. To make these resistance measurements, a precision current source with high compliance is required. The design goals for the current source SCP were that the current source would not limit the input voltage range in a four-wire ohms measurement, and that it would provide 16-bit resolution and the ability to measure 100-ohm resistive temperature devices and 10-kohm thermistors. This required the current source to have ±16V compliance, better than 10-ppm resolution, and greater than 1-gigohm output impedance. For the engineering unit algorithm to work at full speed, the current source has to be calibrated and stable during the measurement cycle.

To aid in conversion of the voltage measurement into resistance, current source values of 488.3 μA and 30.52 μA are used. These sources are programmable to be able to change ranges and turn off the source. Both current sources exhibit low current noise to provide the necessary 16-bit resolution. To put all this capability in a circuit that occupies only 1.5 square inches, we had to design a circuit that used few parts.

Several circuit topologies were tried and discarded for various reasons before the final circuit was designed. The final circuit satisfies all the design goals with a minimum of parts and a simple design (see Fig. 8). It consists of a precision programmable current source driving a high-compliance

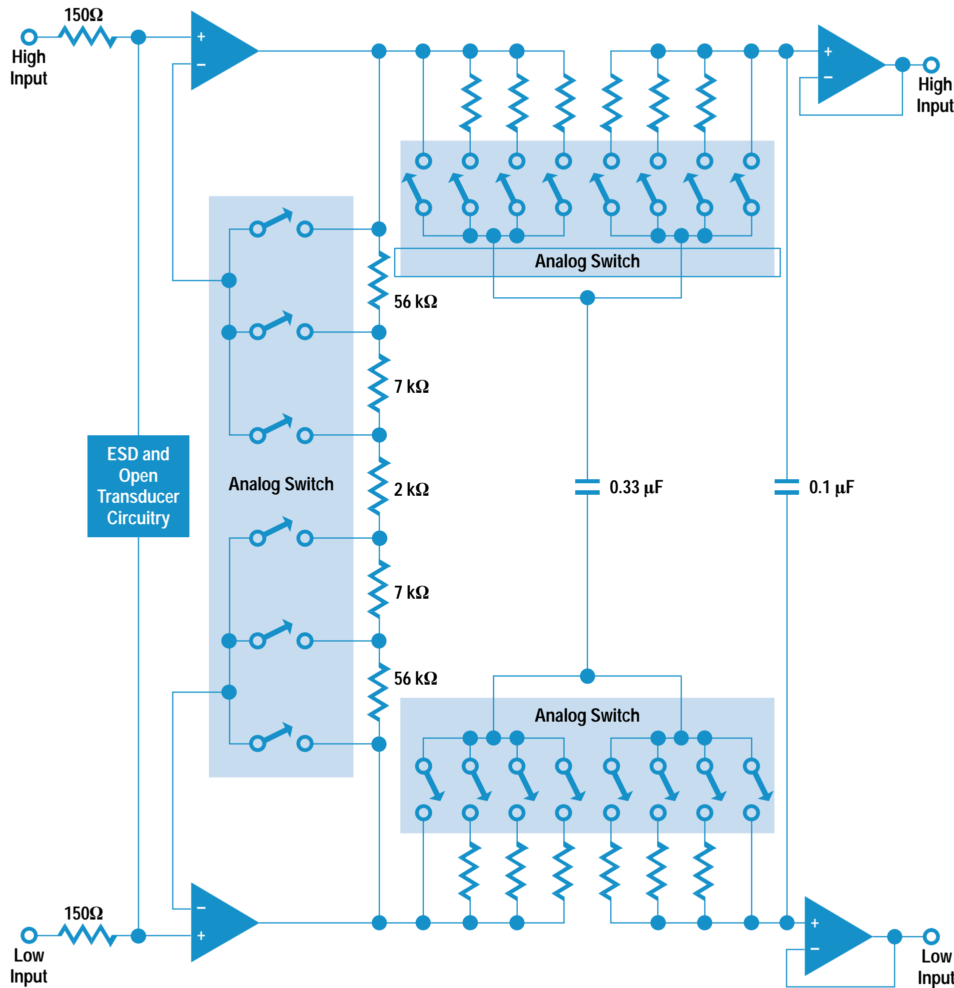


Fig. 7. Programmable gain and filter SCP.

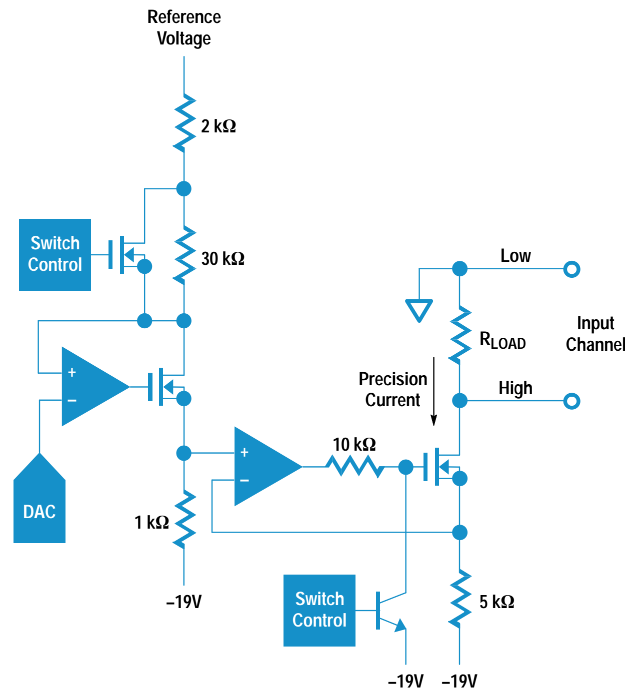


Fig. 8. Current source SCP.

current mirror circuit. The precision dc reference is available to the SCP from the ADC circuits. It is used to produce a precise and stable dc current that is adjusted with a digital-to-analog converter (DAC) and controlled by the operational amplifier. Current switching is performed by shunting the 30-kohm resistor to increase the current. The second part of the circuit reflects and scales this precision current and makes it available to the outside world through the MOSFET transistor. Compliance meets the ±16V goal at extremely high output impedance controlled by the gain of the operational amplifier. Note that this circuit is a current sink rather than a source. This was done to save parts and space.

Strain Completion with High-Accuracy Excitation Source. Customers need to measure strain gauges to understand the stresses and loads on turbines. This measurement is typically made with one or more strain gauges configured in a Wheatstone bridge. An excitation supply is needed for the bridge along with completion resistors for making measurements using only one or two strain gauges.[2] To convert bridge voltage readings into strain gauge values, the bridge excitation voltage must be known or measured. To perform engineering unit conversions to strain gauge values at high speed, the excitation source has to be stable and calibrated to a known value to simplify calculations. The excitation supply must be able to source sufficient current to power all

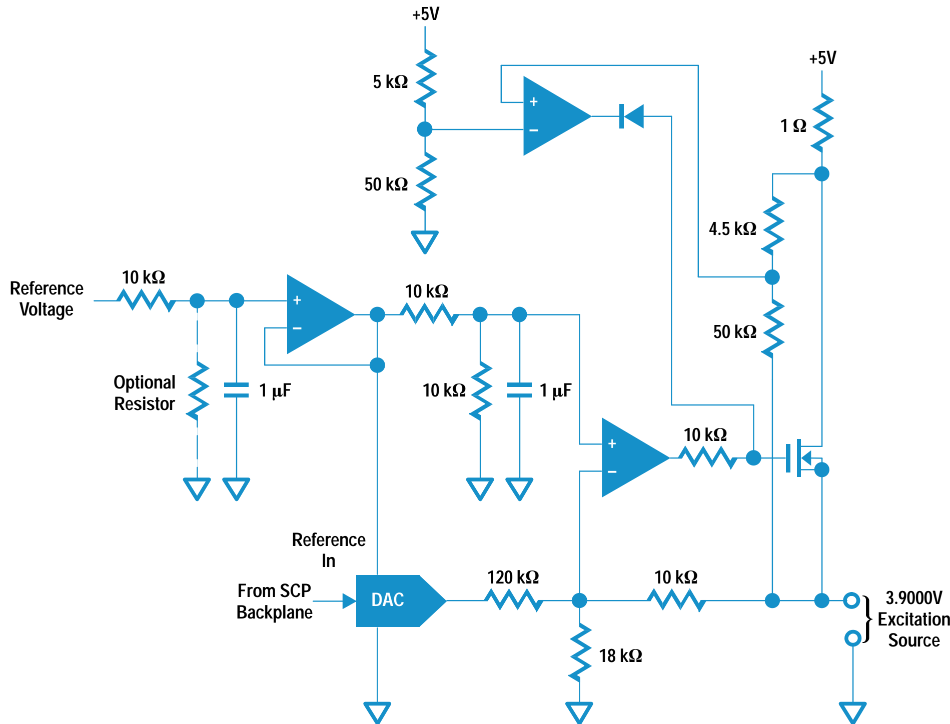


Fig. 9. Excitation supply for the strain gauge SCP.

eight strain bridges on an SCP and be able to survive an accidental short circuit without damage. Since only one excitation source was required for each SCP, space constraints were not as critical.

The circuit chosen provides >400 mA at 3.9000V for excitation (see Fig. 9). The output voltage is calibrated by a DAC to provide this level of resolution. The 3.9000V value was chosen to provide a near full-scale reading on the 4V range. In the event of a short circuit at the output of the excitation source, a fold-back circuit is activated to limit the output current to approximately 75 mA, thus preventing damage to the SCP. Again, the reference voltage provided from the ADC is used as the reference for the excitation supply. It is buffered and then divided down to provide a precise bias voltage for the operational amplifier. A calibration current is sourced at the summing node of the operational amplifier to adjust the output of the excitation source. Note that the calibration DAC uses the same reference as the operational amplifier. This allows the DAC to be used to calibrate any chosen output voltage with the same percentage of calibration span. Pads are provided on the printed circuit board to accommodate different, lower excitation voltages by dividing the reference before buffering. Calibration of the HP E1413 is described in the article on page 25.

When measuring strain gauges, customers often want to verify that the gauges are properly connected by putting a large-value resistor in parallel (shunt) with one resistor or strain gauge in the bridge. The bridge will then become unbalanced by a predictable amount. The bridge voltage can be measured and compared against the expected amount of deflection to determine if the bridge is operating properly. A large-value resistor and a programmable switch are provided on each channel to perform this task. The shunt resistor is usable in any bridge configuration (see Fig. 10).

Other Options

All of the SCPs are designed to accommodate future modifications to the original circuits. Different filter cutoff frequencies, different gains, different output currents, and different excitation source voltages can be accommodated in most cases by simply loading different parts on the printed circuit boards. Pads for other standard-size components are placed on the blank printed circuit boards at strategic nodes to provide for different customer needs with a minimum of additional design time or expense. This has proven to be a popular feature for customers since they can get a custom solution with a minimum of expense and effort.

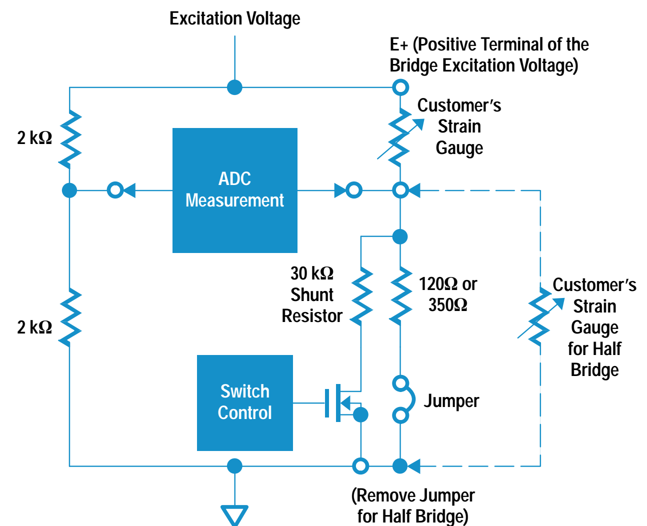


Fig. 10. Shunt resistor for strain gauges.

Conclusion

While the original design goals for the SCPs seemed to be well within reach, the implementation of high-performance circuits capable of providing the needed functionality in the space and at the circuit densities required presented quite a design challenge. High performance was extracted from simple, efficient designs by careful parts choices and placement on the printed circuit boards. Costs were kept low, consistent with the performance required by customers. The results met the original design goals in a way that provides the customer with lasting value.

Acknowledgments

I would like to acknowledge the help of my design colleagues on this project as well as our production engineer, Wes Melander of the Loveland Maufacturing Center, our materials engineer, Beth Delaney, our test engineers, Bert Kolts and Rod Village, and the people in product design services who helped with and designed the printed circuit boards.

Reference

1. R.P. Sallen and E.L. Key, "A Practical Method of Designing RC Active Filters," IRE Transactions on Circuit Theory, Vol. CT-2, March 1955, pp. 74-85.
2. HP Application Note 290-1, *Practical Strain Measurements,* Hewlett-Packard, Part Number 5952-8880, 1981.

High-Throughput Amplifier and Analog-to-Digital Converter

High system throughput in converting analog signals to digital format in the HP E1413 is achieved by not relying on downstream digital processing hardware and software to compensate for analog anomalies and instabilities.

by Ronald J. Riedel

The amplifier and analog-to-digital converter (ADC) section of the HP E1413 provide the interface between the multiplexed analog signals from the signal conditioning pods and the digital world. Some of the functionality provided by this section includes:

- Accepting input signals ranging from a few microvolts to ±16 volts
- Correcting for gain and offset errors on a channel-by-channel basis
- Acquiring each multiplexed signal in turn and settling to full accuracy with no memory of the previous channel, even if the previous channel was severely overloaded
- Converting the analog input to a 16-bit digital number with commensurate linearity and accuracy
- Providing a voltage reference, current source, and calibration voltage source for use by the ADC and the rest of the card.

A key contribution to the overall system throughput and customer ease of use of the HP E1413 is that the above functions are provided smoothly and accurately without the need for further error correction by the downstream digital hardware and software. We resisted the temptation to rely on digital processing to compensate for analog anomalies and instabilities, even though this approach would have saved money and time on the analog design.

Fig. 1 shows the block diagram of the amplifier and ADC section of the HP E1413.

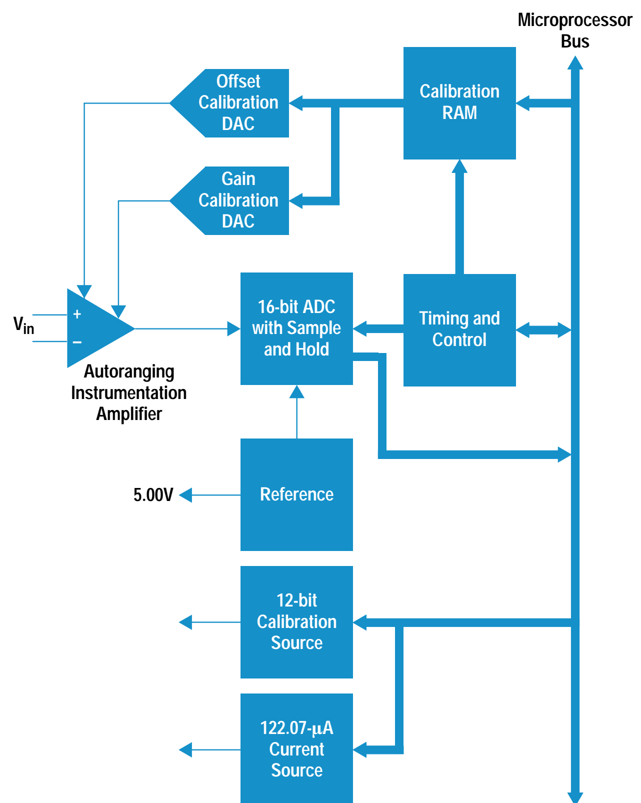


Fig. 1. HP E1413 analog-to-digital converter block diagram.

Amplifier Design and Performance

The performance requirements of a high-resolution, high-speed scanning voltmeter such as the HP E1413 dictate some fairly challenging criteria for the main input amplifier of the ADC section. In fact, more than any other single piece of analog circuitry, the amplifier can define and limit the performance of the entire system. Some of the design requirements for this amplifier included:

- Fast settling time. To achieve full accuracy on a low-level channel following a high-level channel while scanning at 100 kHz, the amplifier must settle to 10 µV after an input of 16 volts, in 10 µs. This represents settling to better than 1 ppm in less than 10 µs.
- Fast overload recovery. So that an overloaded channel does not affect measurements on subsequent channels in the scan list, the amplifier must recover from overload quickly and cleanly. Our goals were recovery to linear operation in less than 1 µs and full settling in less than 10 µs.
- Fully balanced differential inputs. The HP E1413 resolves signals down to 2 µV with a "straight-through" signal conditioning plug-on. However, because of the high conversion rate we cannot use the normal noise reduction techniques of integration, filtering, and averaging. Fully balanced differential inputs are necessary to achieve adequate noise rejection and to eliminate ground loops from the measurement path. The straight-through SCP is described in the article on page 9.
- High common-mode rejection ratio. A high common-mode rejection ratio is necessary for good noise rejection and to allow measurement of sensors such as strain gauges, which typically involve a small differential signal (in the order of

millivolts) impressed onto a large (several volts) dc common-mode signal. Our design goal for common-mode rejection ratio was greater than 120 dB up to 60 Hz on the most sensitive (62.5 mV) voltage range.

- Low noise. Once again, standard noise reduction techniques of averaging, filtering, and integration are not normally available in high-speed scanning applications. Thus, a low-noise amplifier is essential if the full resolution of the HP E1413 is to be usefully realized. Our design goal was a noise level of less than 5 μV rms referred to the input of the amplifier.
- Autoranging with no loss of scanning speed or accuracy. In a scanning system, a customer may connect channel 1 to a thermocouple generating 1 mV and use channel 2 to monitor a 10-volt power supply. The amplifier must be able to range between these two signals at speed, without degrading system accuracy. Also, many customers may be unsure as to the exact voltage expected on a given channel. Autoranging allows them to let the HP E1413 select the optimum measurement range on a sample-by-sample basis. The customer's measurement task is greatly simplified if there is no need to give up speed or accuracy to use autoranging.
- Linearity and accuracy commensurate with a 16-bit system. Our goal was to provide 0.01% overall system accuracy.
- Good dc performance. For all of its high speed, the HP E1413 still has the requirement to be a good dc voltmeter. Thus, low drift (< 10 μV/°C) and low bias currents (< 1 nA) are essential.
- 16-volt differential and common-mode input range. For many applications, particularly in the automotive world, the standard input limits of ±5 volts or ±10 volts seen on many high-speed ADC systems simply are not adequate. We set a goal to provide a full ±16-volt input range so that 12-volt and 14-volt buses can be measured without the use of cumbersome speed and accuracy limiting attenuators.

While none of the performance requirements mentioned above are particularly difficult to achieve in isolation, satisfying all of them simultaneously proved much more challenging. Initially, it seemed to make sense to attempt to implement the amplifier using the classic three-operational-amplifier circuit shown in Fig. 2 with off-the-shelf parts. However, an exhaustive search of manufacturers' catalogs soon revealed that commercially available opamps couldn't do the job. All the parts we considered suffered from one or more of the following limitations: poor settling time or overload recovery, excessive input bias currents, inadequate common-mode rejection ratio, or limited input voltage range.

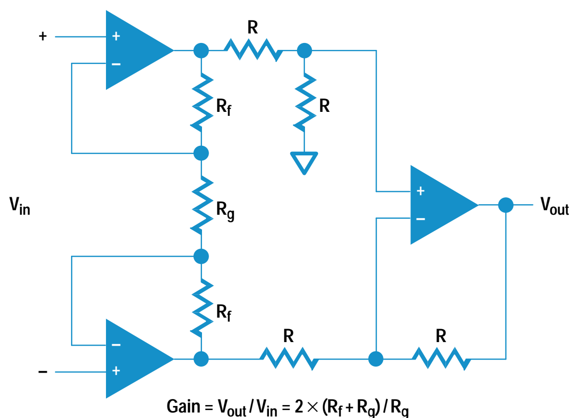


Fig. 2. Classic three-opamp instrumentation amplifier.

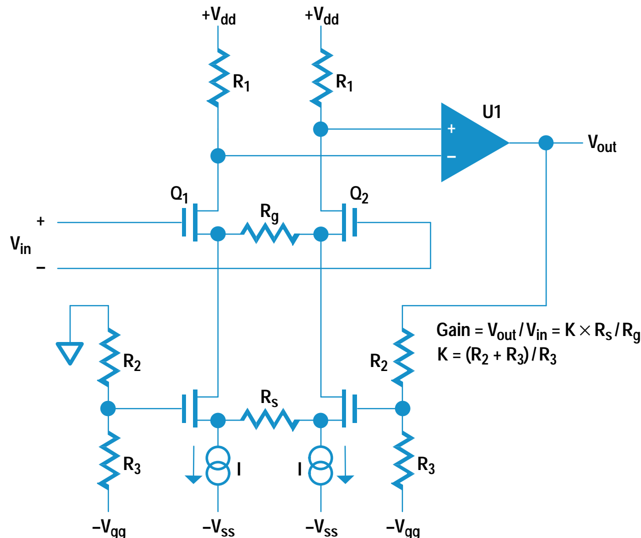


Fig. 3. Basic current-mode instrumentation amplifier.

Another avenue, which involved using a monolithic instrumentation amplifier, was also considered. However, these parts typically require a switch with low resistance and low capacitance such as a relay to switch the gain accurately. This made them useless for meeting our autoranging requirements.

For these reasons, we determined that a discrete amplifier design would be necessary. The amplifier uses a classic current-mode instrumentation configuration, with a special gain switching circuit that is both fast and accurate. It implements overload clamping in such a way as to greatly reduce recovery time by eliminating internal saturation and nonlinear operation. Finally, the amplifier provides for "on-the-fly" gain and offset correction on a per-channel basis so that the ADC sees an accurate signal for conversion. This eliminates downstream time-consuming digital processing to correct the readings for gain and offset errors and allows the full range of the ADC to be used for measurements.

Fig. 3 shows a simplified schematic of the basic current-mode instrumentation amplifier. Because of the feedback action of opamp U1, matched input FETs Q_1 and Q_2 operate at constant, equal currents. Thus, the voltage impressed across R_g is equal to the input voltage with a common-mode shift equal to the V_{gs} of the input FETs. The current through R_g becomes V_{in}/R_g. This current must flow back through R_s to keep the lower current sources satisfied. The output of U1 will servo to make this happen, resulting in an input/output transfer function of:

$$V_{out}/V_{in} = K \times R_s/R_g$$

where

$$K = \frac{R_2 + R_3}{R_3}$$

Thus, the amplifier gain is controlled entirely by the value of K and the ratio of R_s to R_g. K and R_s influence the bandwidth

Binary Ranges Speed Processing

Referring to Fig. 3 in the accompanying article, for the HP E1413 R_s is fixed at 600 ohms. R_g has values of 55.555 ohms, 222.22 ohms, 888.8 ohms, 3.555 kilohms, and 14.222 kilohms, and K = 6.6667. This gives amplifier gains of 72, 18, 4.5, 1.125, and 0.28125. Since full-scale at the ADC is 4.5 volts, this gives full-scale input voltage ranges of 62.5 mV, 250 mV, 1 volt, 4 volts, and 16 volts. At first glance, these full-scale ranges may seem odd; why not choose a more common 1,3,10 or 1,2,5,10 sequence? These values were not chosen for ease of human comprehension, but to interface well to a binary number system. The ranges have full-scale values of 2^n, where n has values of −4, −2, 0, 2, and 4, so the digital reading from the ADC maps directly into the mantissa of an IEEE floating-point number with no further processing required beyond right or left shifts for normalization. This greatly relieves the burden on the downstream processing hardware, freeing it for more productive tasks.

and stability of the amplifier and are normally not varied for a particular design. R_g is used to set the gain because, to a first approximation, bandwidth, settling time, and stability are independent of R_g.

Also notice that ideally the amplifier has no gain for common-mode signals, resulting in a theoretically infinite common-mode rejection ratio. Obviously, there are real-world limitations, which will be discussed later.

Several additions had to be made to this basic amplifier architecture to allow it to meet the performance demands of the HP E1413. The first is a viable means of accurate, high-speed range switching. Simply using several series FET switches to switch in various values of R_g is not workable. The lowest value of R_g used in the HP E1413 is 55 ohms. Any series FET used to switch this resistor in and out of the circuit would have to have an on resistance of much less than one ohm to meet gain accuracy and stability requirements. Such a FET would inherently have a large parasitic capacitance of many hundreds of picofarads, which would destroy amplifier stability, bandwidth, and settling time if used in that area of the circuit.

To meet the performance requirements of the HP E1413, an arrangement of current-steering diodes and small geometry FET switches are used to switch the amplifier gain. Fig. 4 shows a simplified form of this circuit involving two gain ranges. Switches S1a, S1b, S2a, and S2b are small-geometry FETs with channel capacitances around 5 pF and on resistance of 50 to 100 ohms. Since these FETs are in series with the very high impedance of the lower current sources, this range of on resistance has a negligible effect on circuit performance.

As an example of the operation of the circuit shown in Fig. 4 assume that S1a and S1b are on, and S2a and S2b are off. There is then a current path through CR1a and CR1b, enabling R_{g1} to control the gain. R_{g2} is effectively isolated by the back-to-back diodes CR2a and CR2b and the off switches S2a and S2b. CR1a and CR1b operate at constant current, as do Q_1 and Q_2, so that diode nonlinearities have no effect. It is important that the diodes track with temperature since the amplifier input offset voltage is a direct function of the difference between the forward voltage drops of the on diodes.

This gain switching arrangement provides the high switching speed, good settling, and gain stability required for the HP E1413. It also allows us to meet our goal of autoranging at the full 100-kHz scanning speed without compromising accuracy.

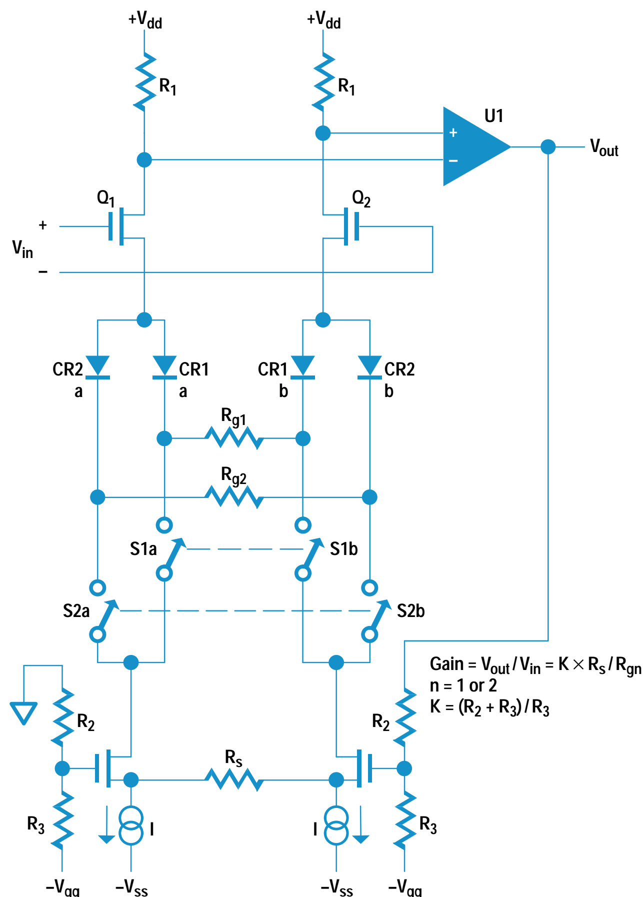


Fig. 4. Basic (2-range) range switching scheme for the current-mode instrumentation amplifier.

For an amplifier to recover rapidly and gracefully from an input overload, it is important that the internal biasing be upset as little as possible when such overloads occur. In the HP E1413, this is accomplished by a special set of clamp circuits that do more than simply limit the input or output voltages. During an overload condition, bias current is simply routed around the gain section instead of through it. This has the effect of reducing the gain to whatever value is necessary to keep the output in the linear region. At the same time, the lower current sources and the output opamp see no change in operating conditions compared to a normal input. Thus, these critical parts of the amplifier are not upset during an overload and recovery is rapid and uneventful (see Fig. 5).

Buffering of the input FETs from common-mode voltage changes is important to achieve the desired high-level common-mode rejection ratio. In the circuit shown in Fig. 3 these FETs will see the entire common-mode voltage as a change in their V_{ds}. Unless the FETs are perfectly matched (an impossibility), this will result in a change of input offset voltage, which will in turn translate into a small differential input signal as the result of the changing common-mode signal.

To prevent this, a cascode arrangement is used (see Fig. 6). The combination of current sources Q_3 and Q_4, along with

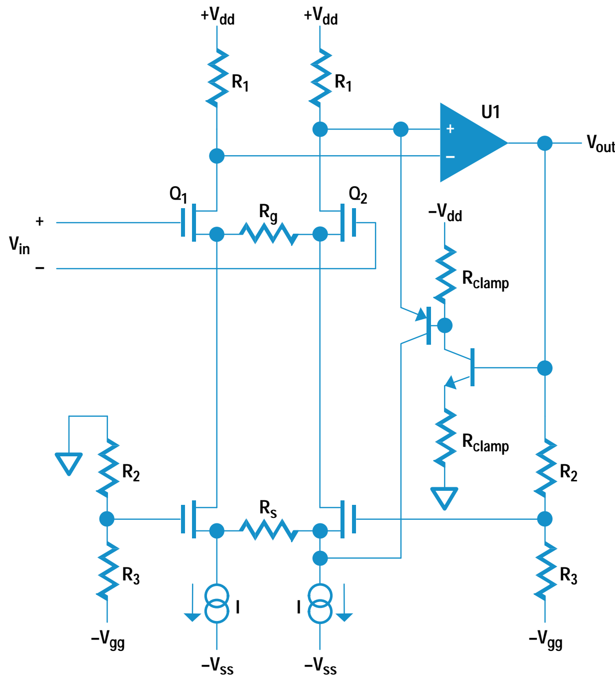


Fig. 5. Amplifier overload clamping technique (only positive clamp is shown).

bipolar pair Q_5 and Q_6, provide a stable, fixed V_{ds} for input FETs Q_1 and Q_2 regardless of the common-mode voltage. Thus, since Q_1 and Q_2 are isolated from common-mode inputs, their offset does not change, and the common-mode rejection ratio remains very high.

The result of all of this careful and sometimes subtle design work is an amplifier that occupies about 2.5 in^2 of printed circuit board area, draws about 300 mW of power, and meets or nearly meets all of the performance criteria outlined above. It is this sort of meticulous design that makes the HP E1413 not just another plug-in ADC card, but a dependable, accurate, information gathering system free from pitfalls and hidden problems for the user.

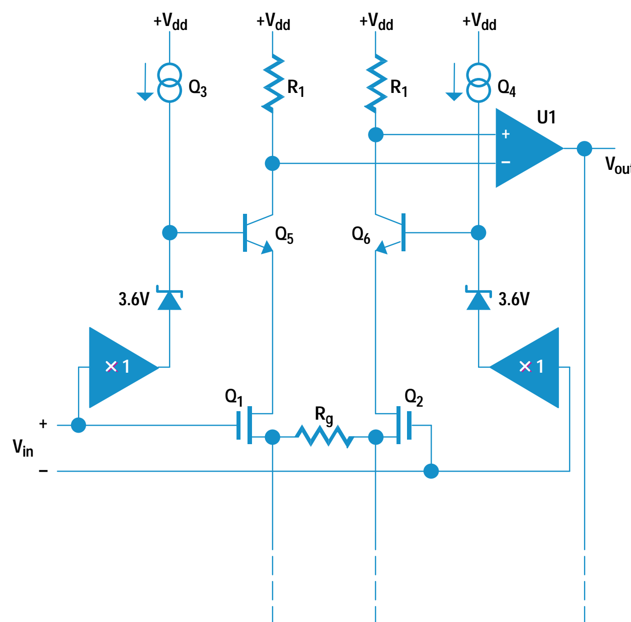


Fig. 6. Upper FET bias scheme for high common-mode rejection ratio.

Calibration, Pipelining, and Timing

On-the-fly gain and offset correction, which is done in hardware in the analog domain, is an important feature of the HP E1413. It may seem old-fashioned to use analog hardware for this purpose in today's world of high-speed number-crunching, but there are some advantages. First, it relieves the downstream digital processing hardware of this task, freeing system resources for more complex tasks such as sensor linearization, engineering unit conversions, and so on.

Secondly, it is useful to recognize that the most fundamental limit on system resolution and dynamic range is the 16-bit analog-to-digital conversion process. Any uncorrected gain and offset errors that occur before the ADC subtract directly from this dynamic range. For example, assume we were doing digital gain correction after the ADC and the uncorrected gain was high by 5%. If a customer tried to measure a 15.9-volt signal in this case the result would be an overrange because 15.9 volts times 1.05 = 16.69 volts, which is above the ADC input range of 16 volts. In fact, for this example, the maximum input voltage that can be measured is only 15.24 volts. This is confusing and frustrating for a customer who presumably ordered and expected to get a 16-volt ADC. Even more confusing, a second unit, presumably identical, might have an uncorrected gain error of 5% low. This unit would then measure up to 16.84 volts before showing an overload.

This kind of customer confusion and uncertainty is unacceptable for high-quality instrumentation. The customer should not have to think about such issues. Analog correction ahead of the ADC process eliminates these issues.

Thirdly, analog offset correction allows us to remove large fixed sensor offsets at the amplifier front end, using the `TARE:CAL` function. This function allows the customer who is only interested in monitoring changes in sensor output rather than absolute value to increase measurement resolution. This is particularly useful with sensors such as strain gauges. Tare calibration is described in more detail in the article on page 25.

Fig. 1 shows the blocks belonging to the gain and offset calibration subsystem. A local RAM stores a separate 12-bit gain correction constant for each of the five amplifier ranges and each of the 64 channels (64 × 5 = 320 gain calibration constants in all). The RAM also stores two offset correction constants for each range and channel (an 8-bit coarse constant and a 12-bit fine constant). This allows us to correct gain errors of up to ±5% and offset errors of up to ±25% of full scale. This relatively large amount of correction range makes the calibration system very robust and able to accommodate a wide variety of signal conditioning plug-ons and sensors. All of these calibration constants are derived automatically during the various system autocalibration functions based on a few fundamental calibration factors measured during factory or calibration laboratory calibration.

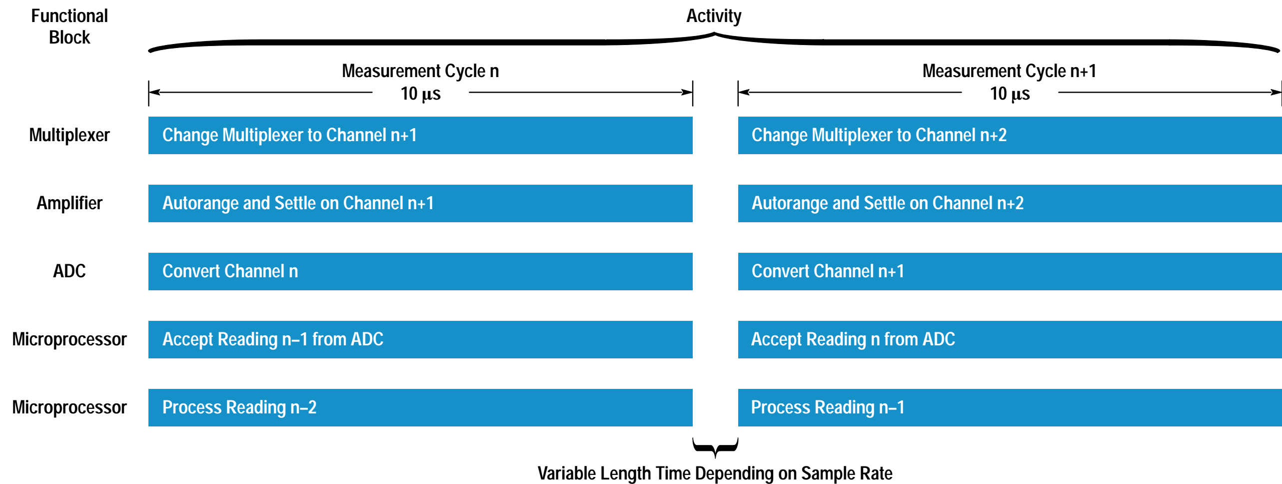


Fig. 7. Timing diagram of an HP 1413 measurement cycle.

Gain and offset correction are interwoven into the measurement cycle such that no speed penalty is incurred. Fig. 7 shows a simplified timing diagram of a typical HP E1413 measurement cycle. An overlapped, or pipelined approach is used so that several operations happen at once.

For example, assume a channel list of eight channels numbered 1 through 8. If at some point during the scan we were to take a snapshot of activity in the HP E1413, we might see the following operations happening simultaneously during a single 10 µs period.

- The multiplexer has switched to channel 6, and the amplifier is autoranging, settling, and applying gain and offset corrections to that channel.
- The sample and hold circuit in the ADC has acquired channel 5, and the ADC is converting this channel to a 16-bit digital word, which will be temporarily stored in the ADC.
- The digital value of channel 4, which has been stored in the ADC since the last measurement cycle, is being transferred to the onboard microprocessor for further processing.
- The onboard microprocessor is processing the reading from channel 3 and transmitting it to the onboard FIFO and current value table* for access by the host computer.

During the next 10-µs measurement cycle, the multiplexer will switch to channel 7 and all other activity will move up one step in the pipeline. This pipelining approach allows the HP E1413 to maintain system throughput at 100,000 readings per second, even though the required operations for a single reading take much longer than 10 µs.

* A current value table is an area of RAM that is accessible to the onboard microprocessor and the host computer. This table stores the most recent reading (current value) for each channel. For monitoring purposes a customer can directly access the most recent readings on any channel without having to sort through possibly hundreds or thousands of readings in the FIFO buffer.

Fortunately, the customer sees none of this complexity. The task of keeping track of all readings and indexing them properly to each channel is taken care of by the onboard microprocessor. This microprocessor also ensures that the pipeline is properly flushed at the beginning and the end of a scan so that no stale readings are transmitted at the beginning of a scan and no good readings are left stranded at a scan's end.

On the HP E1413, a Xilinx FPGA (field-programmable grid array) handles all sequencing and timing of the ADC and amplifier section, including:

- Multiplexer update and channel advance
- Autorange detection and timing
- Calibration RAM interface
- Digital-to-analog converter updates for gain and offset correction
- Start pulse to the ADC.

Using an FPGA for this purpose gave us great flexibility in the design process and reduced the number of required printed circuit board patches and turn-arounds during the prototype phase of the project.

Acknowledgments

Several people deserve thanks for their key roles in this project. They include Don Miller, former section manager, out of whose vision came the whole HP HD2000 effort, Von Campbell, project manager, for his capable management of this project and for keeping us all on track and on schedule, Bert Kolts, test engineer, Vic Anderson, support technician par excellence, and Barb Haas and Anita Helme, who made sure all the parts arrived to build the prototypes on schedule.

On-the-Fly Engineering Units Conversion

An algorithm has been developed that provides engineering units conversion in real time (10 microseconds) in the HP E1413 scanning analog-to-digital converter instrument. The algorithm converts numbers to IEEE 754 standard 32-bit floating-point format.

by Christopher P.J. Kelly

The HP E1413 is a VXIbus 64-channel scanning analog-to-digital converter (ADC) that is used in data acquisition applications. These applications usually measure real-world phenomena such as temperature using a variety of transducers. These transducers (e.g., thermocouples) convert the phenomenon of interest into a voltage, which is connected to a channel of the ADC. The ADC then converts this voltage into a binary digital number that represents the voltage applied to its input. In Fig. 1, for example, the temperature T of a flame is sensed by a thermocouple. The thermoelectric voltage V is applied to the ADC, which generates a digital binary number N1 whose value is determined by V, so its units are typically microvolts. The challenge is to provide a fast engineering units converter to change N1 into a floating-point number N2 in units of temperature such as degrees Celsius.

In the thermocouple example, the relationship between temperature and thermoelectric voltage is described by NIST (the U.S. National Institute of Standards and Technology, formerly National Bureau of Standards) using a high-order polynomial. For a type E thermocouple, for example, NIST models its thermoelectric behavior using a 13th-order polynomial. Considerable computing power is required if this polynomial must be evaluated in real time (10 microseconds in the case of the HP E1413).

Need for Conversion

Why is engineering units conversion important? The reason lies in the nature of the task of continuous on-line data acquisition and the coupling between the instruments and the controller in a VXIbus instrument system.

In continuous on-line data acquisition, measurements are made and recorded continuously for an indeterminate period of time. During this period, data may also be displayed for operators or used for adjustment of the experimental conditions (control). The data should be available in useful and understandable units of measure so that operators and control systems can view and operate upon it easily. Recorded data also should be in a form such that no additional processing is required for the data to be useful, thus helping ensure the correctness of experimental results. For short-duration experiments the data may be buffered in a raw format while a converter slowly massages it into a useful form. But when data is acquired continuously, the acquisition system must be able to convert the data at full speed or eventually the buffers will overflow.

The VXIbus architecture also allows a new, tight coupling between instruments and their controlling CPU. This permits a large increase in system throughput if the instruments are

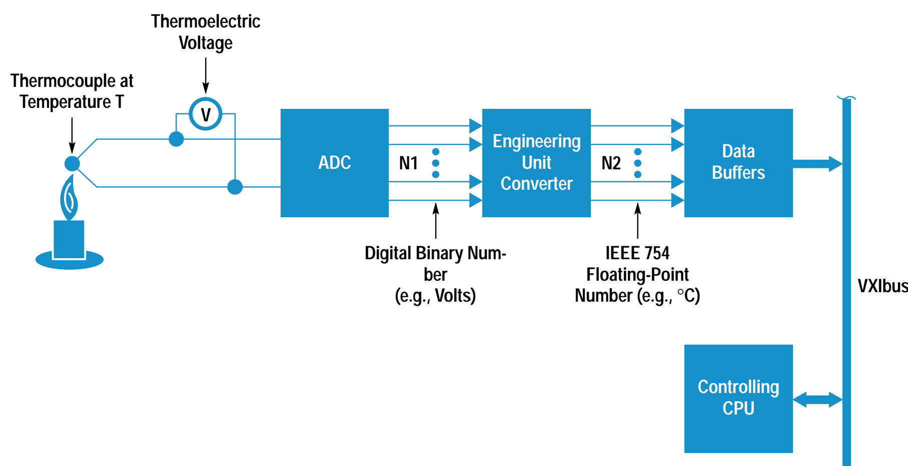


Fig. 1. The components involved in converting a temperature measurement into a floating-point number.

designed to take advantage of the architecture. In some non-VXIbus instrument systems the instruments are coupled to the controller using a communications link, and control or data messages are exchanged by explicit input/output transactions. However, in a VXIbus system the controller can be connected to the VXIbus backplane through its address and data bus, and the instruments' control and data registers appear as memory addresses in the memory map of the VXIbus CPU. In many cases, the CPU actually plugs into the VXIbus cardcage and draws its power from the backplane. In this model, no explicit I/O transactions are required to operate the instruments, so no I/O device drivers or layers of communications protocol are needed, and operations can proceed at full computer backplane speeds.

With this tight coupling between the computer and its instruments, if the instrument is designed to behave like a memory device, the control program can access real-time data at full CPU speeds as if the instrument were part of its memory. If the data is in a computer-native format, the computer can store, display, and manipulate the data immediately, with no format or units conversion. For most of today's computers the IEEE 754 standard floating-point number format is native and is operated upon directly by the floating-point processor in the CPU. If the instrument is able to transfer its measurements in this format, the computer has no burden of translation from a special format to one it can process directly. This means that data can be acquired more rapidly and tighter experimental control can be maintained than with alternative systems of similar cost.

In previous systems, these tasks were often handled by the controlling CPU if the data was required in real time. As data rates rose, considerable CPU power was necessary to do all these tasks in the available time. Some systems used multiple CPUs to handle the load, but the overhead of synchronization and communications among the CPUs also grew to absorb a considerable fraction of the available processing power.

The goal of the HP E1413 engineering units converter is to make it possible to realize the full speed potential of the VXIbus architecture by providing all necessary conversions in the instrument in real time. Each of the 64 input channels requires its own engineering units conversion since transducer types or calibration may vary from one channel to the next. For the HP E1413 this means 100,000 measurements per second are converted to engineering units, formatted according to IEEE 754, and made available to the controller over the VXIbus backplane. 100,000 measurements per second means 10 microseconds per measurement.

Bounding the Problem

Upon first examination of the challenge of on-the-fly engineering units conversion for this product, several factors worked to simplify the problem. First, the ADC only produces a 16-bit binary number as a result of the conversion, meaning that the technique used for the engineering units conversion does not need to exceed 16 bits of resolution. Whatever format is used for reporting results, the measurement itself is intrinsically resolved to one part in 2^{16} (65,536). The second mitigating factor is that although the ADC has five operating voltage ranges, these ranges are related by powers of two. This means that changing ranges only requires shifting the 16 bits from the ADC right or left in an accumulator to maintain correct measurement scaling. Finally, the entire scanning ADC system is calibrated in true volts. This calibration includes multiplexers, filters, attenuators, and gain stages all the way back to the transducer wiring. Every channel is individually calibrated to measure voltage applied to its input terminals. This corrects for any channel-to-channel variation in offset voltages resulting from relay contact thermoelectric voltages and variations in the amplification of the signal-conditioning channel amplifiers. Thus, no postmeasurement correction is necessary, allowing the measurement to proceed at maximum speed.

This third factor means that all engineering units conversion coefficients can be calculated in advance, referenced to absolute voltage. If the system were calibrated in some arbitrary ADC voltage units, we would require dynamic computation of coefficients based on the state of calibration of each individual ADC, a production engineering nightmare.

Efforts have previously been made to speed up engineering units conversion in similar systems, and a variety of shortcut techniques have been developed. These techniques generally trade off accuracy, speed, and memory table size, and each technique has strengths and weaknesses. These techniques include development of low-order polynomials to approximate the NIST equations with fewer terms and hybrid lookup-table-with-correction-factor techniques. In the limit, it is possible to perform simple table lookup, with the ADC reading being used as an address index into an array of results. This technique is very fast, but uses a lot of memory for the lookup tables. In the case of the E1413, with 16 bits of resolution and 64 separate channels, this technique would have required 16 megabytes (64 channels × 16 bits of resolution × 4 bytes per result = 64 × 65536 × 4 = 16 megabytes) of tables, too expensive a solution. On the whole, we needed a technique that would precompute as much of the problem as possible, leaving little remaining work to be done in real time.

New Algorithm

The HP E1413 engineering units conversion algorithm uses a linear approximation technique to convert the ADC binary voltage numbers into engineering units. The technique divides the operating voltage range of the ADC into a number of segments of equal size and fits a straight line to the data in each segment. The number of segments can be adjusted to provide an acceptable level of conversion error, with more segments yielding lower error. In our case, we found that 128 segments were sufficient to limit errors to acceptable limits.

In Fig. 2, the algorithm is represented schematically. On the left, a digital binary number from the ADC is divided into two sections. The high-order bits are used to address a table of engineering units conversion coefficients. The low-order bits enter the numeric processing pipeline and are multiplied by the slope coefficient for this table segment, resulting in the product (P). This product is then added to the offset coefficient for this table segment, which represents the value of the engineering units conversion equation at the beginning of the segment. The result (R_0) is now in engineering units, and in a simple model can be used as output. This process of lookup, multiply, and add can be accomplished

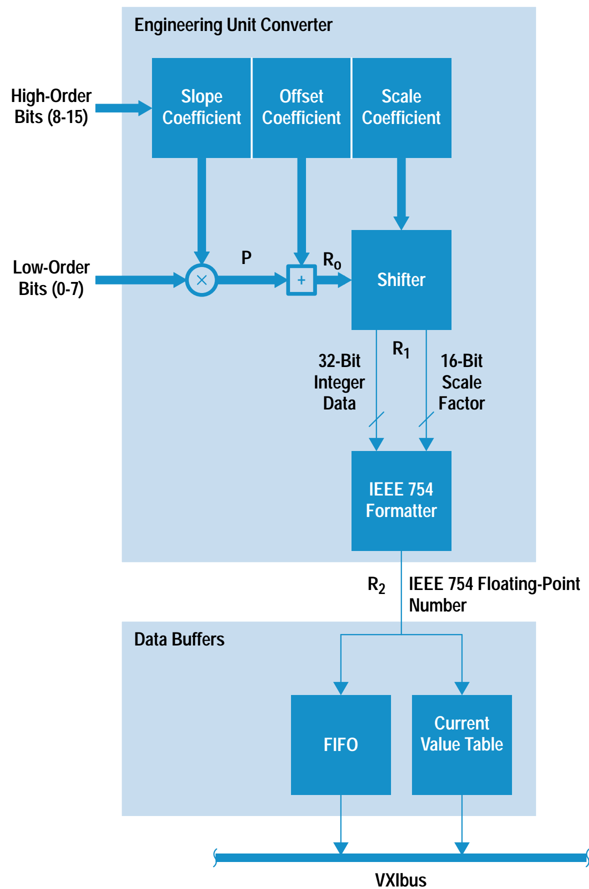


Fig. 2. A block diagram of the engineering unit conversion process.

very quickly compared to evaluation of a high-order polynomial. In fact, the process is simple enough to be performed entirely in a simple hardware state machine consisting of ROM, multiplier, adder, and clock control devices.

In the HP E1413, additional steps are added to keep the resolution of the conversion up to an acceptable level. In general, an engineering units conversion equation will have relatively high slope coefficients in some parts of the curve and relatively low slope coefficients in others. The comparative sizes of the slope and the offset will also change over the width of the conversion curve. To keep the resolution of the conversion process up to acceptable levels when using integer arithmetic, the HP E1413 conversion also includes a third, scaling, coefficient. This factor is used to adjust the scale of the slope and offset coefficients, and is corrected for in a postconversion rescaling (right or left shift as appropriate). The scaled result (R_1) is then reformatted according to IEEE 754 before the result (R_2) is presented to the controlling CPU.

To show graphically how the algorithm operates, Fig. 3 shows the thermoelectric voltage curve for a hypothetical temperature transducer. On the vertical axis we see temperature and on the horizontal axis we see the resulting voltage. The curve illustrates a nonlinear relationship between temperature and voltage. When the ADC measures the transducer, it puts out a digital binary number V_{meas}. When this voltage is presented to the engineering units converter, it is found to fall into voltage segment n, because its voltage lies between the beginning voltage of the segment (V_n) and the ending voltage of the segment (V_{n+1}). When the engineering units

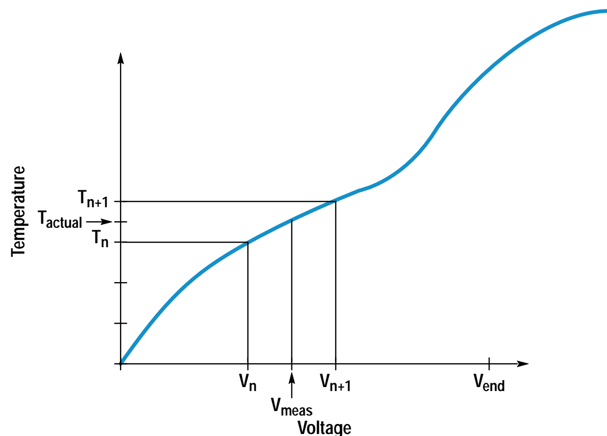


Fig. 3. Thermoelectric voltage curve for a hypothetical temperature transducer.

converter receives V_{meas}, it divides it into high-order bits, which have a value of n, and low-order bits, which indicate how far into the V_n segment the actual voltage lies.

The engineering units converter next uses the high-order bits to look up the nth segment slope coefficient from the conversion table. The slope coefficient is multiplied by the low-order bits, whose value is ($V_{meas} - V_n$). The product of this multiply has a value of ($T_{actual} - T_n$) on the vertical axis. That is to say, the product is equal to the distance above the base temperature of the segment where the true temperature lies. The second step of engineering units conversion is to fetch the offset coefficient (T_n) from the coefficients lookup table and add it to the product of the multiply. The result of this addition is now in units of temperature. This value is now scaled and converted to floating-point number format, and is available for use by the controlling CPU.

Results

The multiply-add sequence is a core function in most digital signal processing (DSP) algorithms. It is a function optimized by most DSP processor chips into a very fast operation. This makes the architecture of DSP processors well-suited to the engineering units conversion algorithm, for which multiplication, addition, and speed are all important. The HP E1413 uses a Texas Instruments TMS320C51 DSP processor chip as its onboard CPU, and is able to execute this algorithm in the available 10 microseconds, along with other functions such as measurement sequence control. The processor also handles measurement trigger counting and timing functions during measurement sequences and calibration and measurement setup commands when the instrument is not actively acquiring data.

As mentioned above, the algorithm can be adjusted to trade conversion accuracy for coefficient table size. In the HP E1413, the algorithm divides the engineering units space into 128 segments, which yields 512-word conversion tables for each transducer. Since the thermocouple is one of the target transducers for this product, it is instructive to evaluate the errors generated by the algorithm when using thermocouples. Fig. 4 shows a graph of the algorithm errors when the transducer is a Type E thermocouple. The largest errors occur near 0°C and are nearly 0.05°C. Over most of the temperature range the errors are below 0.005°C, far below the ASTM

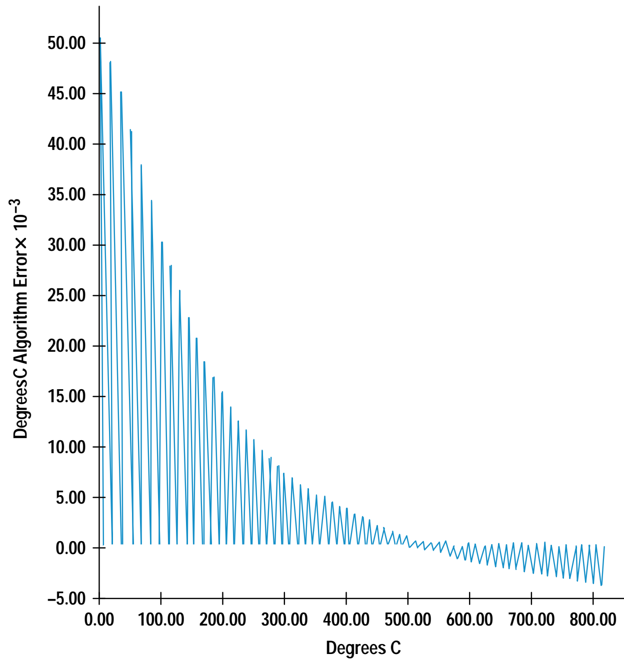


Fig. 4. Engineering unit accuracy with a Type E thermocouple.

manufacturing error specification for the thermocouple wire itself, which rises to ±4.4°C above 800°C.

The shape of the error curve is determined by the rate of change of the thermoelectric voltage for the thermocouple over temperature. The thermoelectric voltage change per degree of temperature change is called the Seebeck coefficient of the thermocouple. As the Seebeck coefficient changes, the algorithm must approximate a curving line segment with a straight line. This leads to a small error in the middle of the line segment, with almost zero error at the endpoints. The greater the curvature of the function being approximated, the greater the error. The errors generated by the conversion algorithm are far below the measurement errors of the analog hardware or those of the transducers themselves.

Summary

The new segmented linear approximation algorithm allows real-time conversion of analog measurements into engineering units. The algorithm provides ample conversion accuracy for the target hardware and transducers, and executes in less than ten microseconds using the microprocessor in the instrument. Performing this conversion in the instrument allows the complete measurement system to make usable measurements faster and at lower system cost than was possible with previous instruments.

Acknowledgments

Thanks to the whole HP E1413 design team for their cooperative can-do attitude throughout this project. This project has been a challenging and enjoyable experience, and I have been privileged to work with such fine people.

Built-In Self-Test and Calibration for a Scanning Analog-to-Digital Converter

Onboard calibration capability enables the HP E1413 to calibrate all 64 input channels in under 15 minutes, many times faster than the manual calibration techniques previously required in similar systems.

by Gerald I. Raak and Christopher P.J. Kelly

The HP E1413 is a 64-channel scanning analog-to-digital converter (ADC) with configurable signal conditioning plug-on (SCP) circuits. The HP E1413 combines signal conditioning, analog multiplexing, analog-to-digital conversion, and digital data processing in one integrated subsystem. Similar subsystems are made up of separate components, often combined by the end user. In such systems it is difficult for the designer of one component to provide end-to-end calibration and self-test since the designer has no control over the design of some components. Final integration is often left to a system integrator or to the end user, who must then add hardware and software engineering to make the system fully functional. Calibration of high-channel-count data acquisition systems of this sort may take many hours or even days, which drives up the cost of ownership.

With design control of all four major components, the HP E1413 design team was able to add considerable value by integrating these components and their interaction during calibration, self-test, and measurement operations. This is accomplished using additional calibration hardware and software that enable stimulus and measurement access to all major components of the measurement hardware.

Calibration and Self-Test Hardware

Three major hardware components were added to the HP E1413 to enable self calibration and test: onboard voltage and resistance reference sources, a calibration bus that connects sources to various points in the circuit, and multiplexers and relays to control measurement and source signal routing.

Fig. 1 shows a simplified diagram of the HP E1413 calibration hardware. The voltage and resistance source section provides a precision 7500-ohm resistor, a programmable voltage source, and a short circuit (zero voltage) reference. The calibration bus (CALBUS) connects the various sources either to the input of the signal conditioning plug-on boards or to an external multimeter via the front-panel connector. The 64 calibration relays serve two purposes: first, to allow calibration sources to enter the signal conditioning of a channel, and second, to isolate the HP E1413 input stages from the customer wiring when power is lost or when excessive input voltages are detected. The semiconductor multiplexers MUX B and MUX C allow the reference sources to be connected to the calibration bus or directly into the ADC input amplifier stage.

Different Types of Calibration

The objectives of the HP 1413's built-in calibration are to provide high measurement accuracy and to make calibration simple and fast for customers, including measurement systems with more than one HP E1413. The calibration tree in Fig. 2 shows that there are several types of calibration available for the HP E1413. The three main types are calibrations using an external standard multimeter, working calibrations including those done using the multimeter, and internal self-calibrations. Fig. 3 shows the HP E1413 components calibrated by these different calibration types.

Working calibrations and internal calibrations are depicted in Fig. 3 as nested boxes because these calibrations are related in a hierarchical manner. Subsets of these major calibrations are accessible to the customer because their speed and convenience make them individually useful (see "A Hierarchy of Calibration Commands," on page 28). For best results, internal self-calibrations and working calibrations should be performed in the HP E1413 working environment. This means the HP E1413 should be warmed to normal operating temperature with normal operating airflow from the VXIbus cardcage. There can be significant temperature and airflow differences inside a VXIbus mainframe between a standards laboratory environment and the working environment depending on ambient temperatures and how many cards are loaded into the VXIbus mainframe. An external multimeter with its heated reference and internally controlled airflow can handle these environment changes better than an HP E1413.

As the first stage of calibration, the external transfer standard digital multimeter should be calibrated in a standards laboratory. This multimeter should be a 6½-digit (21-bit) or better multimeter (such as the HP E1410 or the HP 3458A digital multimeters). The external multimeter is then taken to the HP E1413 operating environment where it can be used to calibrate multiple 16-bit HP E1413s. This makes the standards laboratory's calibration easier, and an HP E1413 calibrated in its working environment will be more accurate in that environment. In some systems, the digital multimeter is dedicated to the test system and wired to the HP E1413 calibration bus connectors and is only removed when the multimeter requires recalibration.

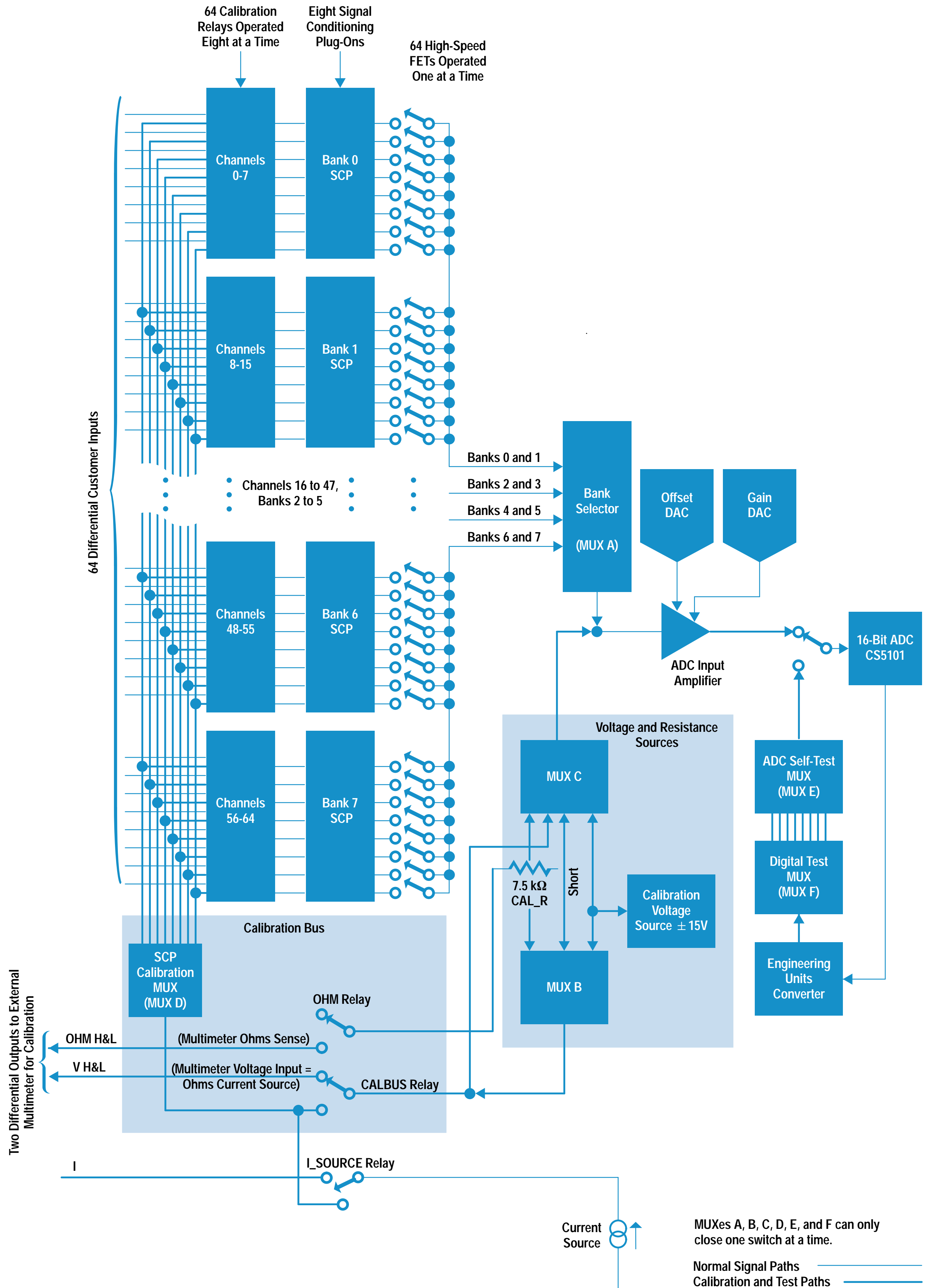


Fig. 1. A block diagram of the HP E1413 circuits used for calibration and test.

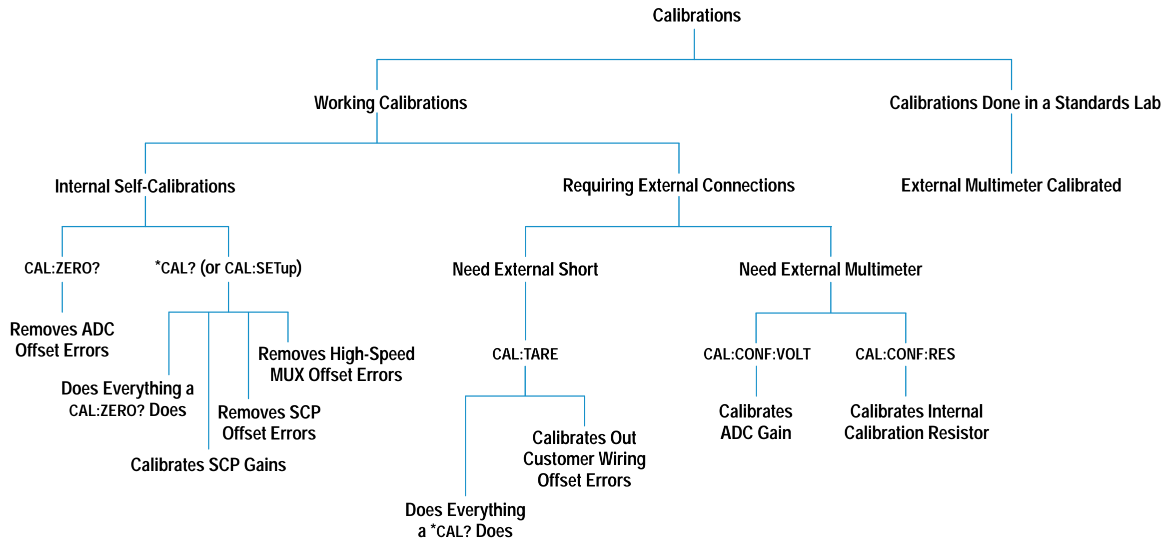


Fig. 2. HP E1413 calibration tree.

External HP E1413 Working Calibrations

External working calibration of the HP 1413 requires a multimeter to establish ADC gain calibration and the exact value of the precision resistor in the HP E1413. The ADC gain calibration forces the gain of the HP E1413 ADC to match the gain of the external multimeter on each of the five measurement ranges of the HP E1413. This calibration takes less than five minutes using an external controller sending a sequence of setup and measure commands for each measurement range.

In the following discussion it is assumed that the multimeter is an HP 3458A DMM or other HP bench-type multimeter. Other types of multimeters may require different connections.

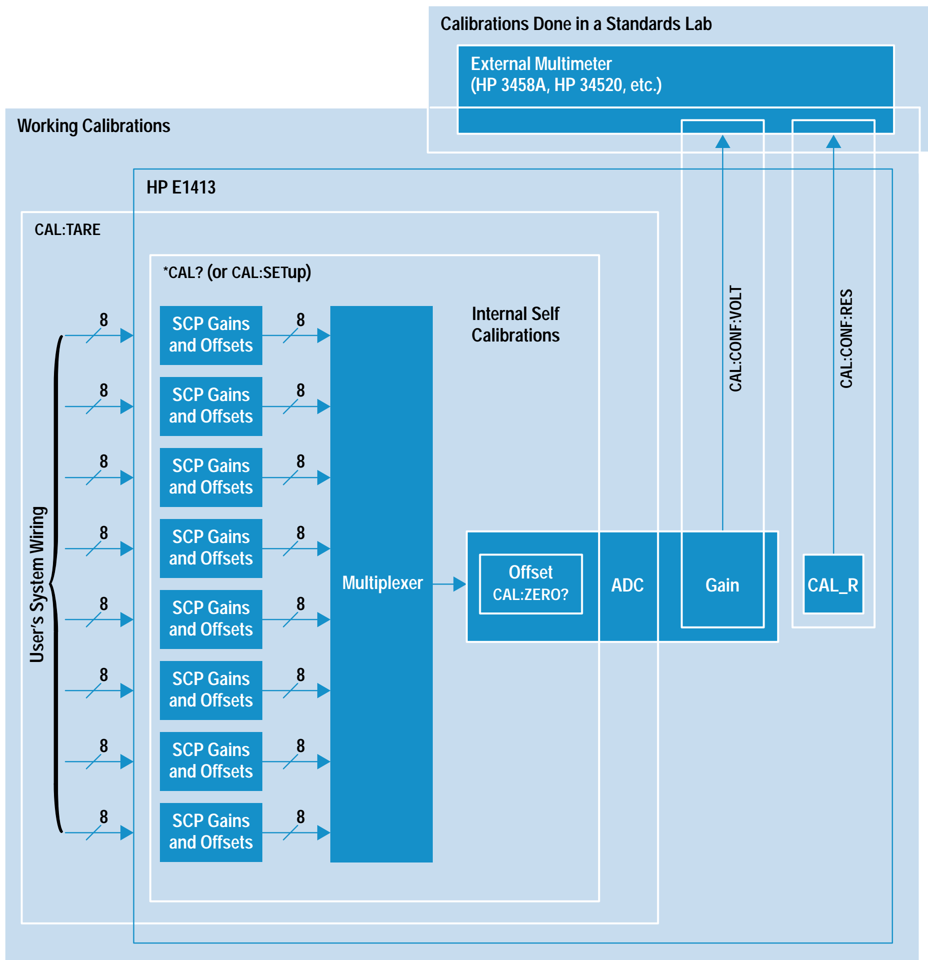


Fig. 3. Areas of the HP E1413 affected by the different calibration types.

A Hierarchy of Calibration Commands

Calibration of the HP E1413 ADC module can be performed using only one external instrument, a digital multimeter. As implied in Fig. 2 in the accompanying article there is a hierarchy of calibration types in the HP E1413 that can be invoked by the calibration commands.

At the highest level are calibration commands that evaluate the HP E1413 transfer-standard components. These commands, which include CALibrate:CONFigure:VOLTage and CALibrate:CONFigure:RESistance, allow an external digital multimeter to make measurements to establish the value of key parameters in the HP E1413 circuitry. These key parameters are the ADC input amplifier gain on each of five ranges and the value of the precision resistor in the HP E1413. All other HP E1413 calibrations are based on these six parameters.

Below these calibrations are the working calibrations,which include CAL:TARE, *CAL?, and CAL:ZERO?. These calibrations affect smaller and smaller parts of the HP E1413 and are shown in Fig. 2 as being entirely contained within the previous type of calibration. The innermost (smallest span) calibration is CALibrate:ZERO?. CAL:ZERO? is used only to correct the ADC for drift in voltage offset, which usually results from temperature change. This command is very fast (less than five seconds) and can be executed frequently to compensate for any changes in the temperature or flow rate of the air cooling the HP E1413.

The *CAL? command is used to correct for any voltage offset caused by the SCPs or the 64-channel multiplexer, and for any gain errors in the SCPs. For example, if channel 1 of a SCP has a nominal gain of eight but an actual gain of 7.998, the *CAL? command will correct for the difference by adjusting the ADC input amplifier gain upward by a factor of 1.00025 whenever channel 1 is being measured. Every channel containing an input SCP is calibrated this way during the execution of the *CAL? command. Since up to 64 channels are calibrated, the *CAL? command can be time-consuming, usually requiring 5 minutes or more to complete. The onboard current source and any current or voltage sources on the SCPs are also calibrated during *CAL?. Fig. 3 in the accompanying article shows that CAL:ZERO? is also performed by *CAL?.

The outermost layer of calibration is CAL:TARE which adjusts all measurement channels for any voltage induced by customer wiring. To understand the tare calibration, consider an analogy from weight measurement. A customer wishing to buy rice by weight places the rice into a plastic container on the scale. The customer does not wish to pay for the weight of the plastic container in addition to the rice. So the container is first weighed alone (this is the tare weight), then its weight is subtracted from the total weight of the container and rice. In some measurement systems, the wiring between the HP E1413 and the transducer may cause some small voltage to be added to the actual transducer output voltage. Any measurement of transducer voltage will include this small additional voltage caused by the wiring. Therefore, the wiring is short circuited at the transducer, and the tare voltage is measured for each channel. This value is used during *CAL? to adjust calibration so that the tare voltage is subtracted from all future measurements on that channel. Again, as Fig. 3 shows, *CAL? and CAL:ZERO? calibrations are performed by CAL:TARE.

At the start of the calibration the HP E1413 is commanded to put out a voltage from its voltage source to the external calibration connector (connection path = calibration source to MUX B to the CALBUS relay to V H&L to the multimeter voltage input).†

Next, the HP 3458A multimeter reads the voltage and passes the reading back to the HP E1413. The HP E1413 calibration firmware now connects the source to the ADC (connection path = calibration source to MUX C to the ADC input amplifier to the ADC) and adjusts the input amplifier's gain to match the value of the reading from the external voltmeter. The gain of the HP E1413 ADC can only be calibrated in this manner, so the stability of this gain is important to measurement accuracy.

The precision resistor is also measured by an HP 3458A multimeter using a very similar setup and measure sequence (connection path = multimeter ohms current source (voltage input) to V H&L to the CALBUS relay to MUX B to CAL_R and multimeter ohms sense input to OHM H&L to the OHM relay to CAL_R). The resistance measurement made by the external multimeter is passed to the HP E1413 and stored for use during internal self-calibration.

For best accuracy before an important data run, the ADC gain and the precision resistor can be recalibrated in less than five minutes if the external voltmeter is left connected (connection path = HP 3458A voltage input/ohms current source to V H&L and HP 3458A ohms sense input to OHM H&L). Multiple HP E1413s can be permanently wired in parallel, but only one HP E1413 can be doing these calibrations at a time. If an HP E1413 is not doing these calibrations, it will disconnect itself from the external multimeter by opening the OHM and CALBUS relays.

Internal HP E1413 Calibrations

The first internal calibration removes the offset errors in the ADC input amplifier with the command CAL:ZERO?. This calibration, which is used to remove any offset error caused by a temperature change in the ADC input amplifier, is entirely self-contained and requires no external equipment. It is performed by the onboard microprocessor by connecting the ADC input to a reference short and adjusting the ADC offset DACs to obtain a zero reading on each of the five ADC voltage ranges. This is similar to a multimeter's autozero function except that it is not done automatically, but on command. The CAL:ZERO? calibration takes less than five seconds to run and should be executed frequently (every five minutes if possible).

The second internal calibration is invoked with the SCPI *CAL? command and is a complete internal self-calibration that removes errors between the customer inputs and the ADC chip. These errors include SCP gain errors and any offset caused by the SCPs or the 64-channel multiplexer. This internal calibration also adjusts the value of voltage or current supplied by the SCPs to external transducers. The 7500-ohm internal calibration resistor is used to convert the HP E1413's internal current source and the SCP current sources to a voltage that is measured and calibrated by the internal ADC.

The *CAL? function measures and corrects HP E1413 offset and errors on a channel by channel basis. For offset calibration, the SCP inputs are connected one at a time to the same internal short used during a tare calibration (described below) (connection path = one bank of calibration relays to MUX D to CALBUS relay to MUX B to short). Next, the ADC output is forced to the value of the voltage difference between the internal short and the customer supplied short stored during the tare calibration by adjusting the value of the ADC offset DAC.

For gain calibrations, a measured input voltage is applied to each channel of the SCPs (connection path = calibration source to MUX C to ADC input, then calibration source to

† Refer to Fig. 1 for the components given in these calibration connections.

MUX B to CALBUS relay to MUX D to a bank of calibration relays to the associated SCP and finally to the ADC input amplifier). The output voltage of the ADC input amplifier is now adjusted to make it match the gain of the SCP channel. The *CAL? should be done immediately before an important test and any time there has been a significant temperature change. The driver allows multiple HP E1413's to be calibrated with the *CAL? command at the same time. When the HP E1413 contains SCPs with slow filters, no external voltage should be applied to the input terminals during *CAL?.

A tare calibration measures and corrects for the customer's and HP E1413's offset errors on a channel by channel basis. The offset errors this calibration removes come from the SCPs, the ADC input amplifier, and the ADC. There are 64 offset calibration constants that represent the voltage difference between an internal short (connection path = one bank of calibration relays to MUX D to CALBUS relay to MUX B to short) and the customer-supplied short (connection path = customer input to a calibration relay to an SCP to an FET switch to MUX A to the ADC input). This is the normal connection path when an HP E1413 is connected to measure customer transducers. When measuring the customer short, the measurement period is set to reject power line noise. Tare calibration requires from 5 to over 30 minutes depending on the SCPs that are installed, since the SCP filters must settle for external measurements. Each time CAL:TARE is executed, a full set of *CAL? constants must be generated. The SCPI driver allows multiple HP E1413's to be tare calibrated at the same time because of the length of time required. Tare calibration is usually a one-time calibration that is done the first time customer wiring is attached. However, it should be performed whenever the customer wiring offset changes.

Analog Self-Test

Analog self-test uses the same hardware as calibration. Three sections are associated with analog self-tests. The first section is centered around a digital multiplexer (MUX F in Fig. 1). Eight tests are used to ensure that the control signals from the onboard microprocessor are correctly getting to the ADC, relay drivers, and multiplexer drivers. The second section is centered around the analog multiplexer in the ADC (MUX E in Fig. 1). For this section, 24 tests are used to ensure that the analog portion of the ADC is working. The third and final section uses the calibration hardware to connect to the calibration paths and make sure the offsets and gains are within the three-sigma limits of the hardware. The third section also tests the SCPs. There are about 14 tests in this section, some of which are run on each channel. The actual number of tests and which tests are run depends on the type of SCPs installed. This section of self-test is the most time-consuming because some of the tests have to be run on each of the 64 channels. As with the calibrations, if SCPs with slow filters are installed the self-test will take longer to let the voltages settle before taking measurements. If only HP E1413 Option 11 SCPs (with no filters) are installed, the self-test will take less than six minutes. If eight HP E1413 Option 12 SCPs (with 10-Hz filters) are installed, the self-test can take over 30 minutes. The self-test should be run only after the HP E1413 has been allowed to warm up in its operating environment. An HP E1413 may fail self-test during the first five minutes after power-up because test measurements may drift out of limits as the circuitry warms up.

Conclusion

By adding a modest amount of circuitry and firmware to the HP E1413, the speed, simplicity, and accuracy of calibration were improved. The circuitry includes relays and analog multiplexers to route calibration signals to different sections of the instrument. The only external instrument required for calibration is a digital multimeter, and most calibration functions are performed in an entirely self-contained manner. The same circuitry used for calibration is also used for self-test. During self-test various sections of the instrument are evaluated and diagnostic functions report failures to a very fine degree.

Manufacturing Test Optimization for VXI-Based Scanning Analog-to-Digital Converters

The high density of the hardware for the HP E1413 scanning analog-to-digital converter, the low cost per channel, and the wide variety of optional signal conditioning plug-ons require a production test strategy that is fast, flexible, and efficient.

by Bertram S. Kolts and Rodney K. Village

The wide range of features and functionality available in the HP E1413 scanning analog-to-digital converter, the flexibility provided by the signal conditioning plug-ons (SCPs), and the desire to occupy only a single VXIbus slot dictated a design that is complex and requires a high-density hardware implementation using double-sided surface mount technology with numerous fine-pitch components. Since the cost per measurement channel is aimed at being one of the lowest in the industry, the challenge to manufacturing was to develop prototype and production test and evaluation strategies that would be cost-effective.

Four project test goals were established at the start of the project to provide a low-cost test solution. The first goal mandated the use of existing test systems to reduce hardware development costs. The second goal was to reduce the cost of testing by removing defects as early in the production process as possible. The third goal was to reduce unit test times by a factor of three relative to similar products currently in production. The fourth goal was directed specifically at the SCPs. Because the SCPs have high production volumes (eight SCPs for each HP E1413 motherboard) compared to the HP E1413 motherboard and because of their low cost and relatively straightforward hardware design, the goal was established that the plug-ons would only be tested at one point in the production process.

Test Strategy

From these four goals the test strategy for the HP E1413 was developed. The need to remove process defects as early as possible in the production cycle requires testing the product at the completion of circuit board loading in the surface mount fabrication area (see Fig. 1). Testing at this point ensures that any process defects are removed as early in the manufacturing process as possible so that the product can function and be capable of being calibrated at final test. This test strategy enabled us to recognize another goal for the project and that is to refrain from duplicating tests in the surface mount and final test areas.

“Just-enough-test” is another test strategy that was implemented for the HP E1413. The aim of just-enough-test is to try to minimize test costs while at the same time ensuring that product quality is maintained at a high level. The simplest way to ensure high product quality is to test all of a product’s specified parameters. Although this is certainly a safe and easy approach, it is also the least cost-effective in terms of development time and production test time. On the other hand, the lower the amount of test coverage the greater the risk of shipping a defect. Most test engineers probably tend to err on the side of too much testing, rather than too little, to ensure that as many defects as possible are found. With the HP E1413 we wanted to optimize the amount of testing. The HP E1413 test suites use several techniques to address this issue.

Reducing test costs requires not only that test run times be kept low, but that setup times be held to a minimum as well. To accomplish this, especially in the surface mount area, it was necessary to minimize the hookup time by requiring

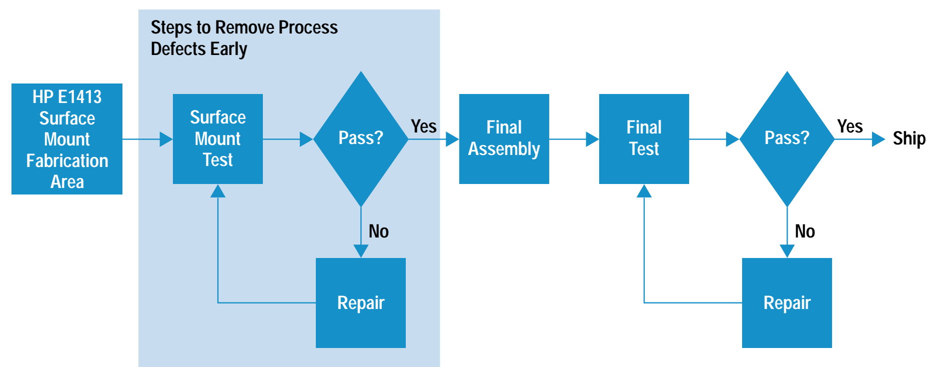


Fig. 1. The HP E1413 assemblies pass through a two-stage test and repair process. The surface mount test stage is aimed at detecting process defects and the final test stage is intended for calibration and performance verification.

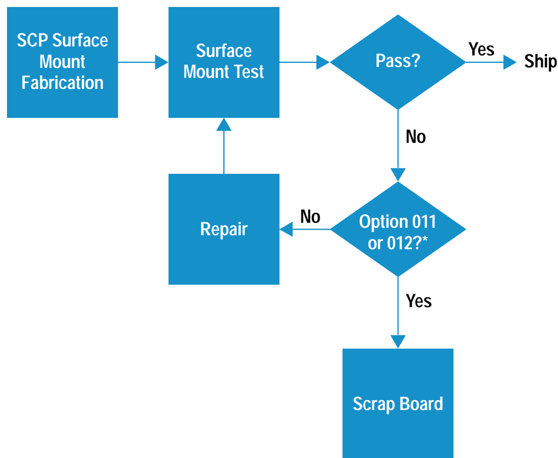


Fig. 2. The test and repair process for SCPs. Signal conditioning plug-ons are tested only after fabrication because they do not require final assembly or calibration.

front-panel connections only. To this end we needed access to major functional blocks within the product from either the backplane or the front-panel user interface. We decided that an internal analog self-test multiplexer that provides access to the internal functional blocks and is accessible from the front panel would be the best solution.

While the goal of doing a single manufacturing test for the signal conditioning plug-ons was easily met, the test process for the plug-ons still required a test set with a minimum amount of components and the shortest possible repair times. An analysis of the cost of repair versus the factory cost of various SCPs showed that for the lowest-cost boards that did not pass the tests, it was more economical to discard such boards rather than repair them (see Fig. 2). This was a major change in our old manufacturing paradigm in which all failures had to be repaired.

Implementation

Having defined the test goals and strategies and adding the goal that tests would not be duplicated from one area to another, we were able to split the test development workload between two test engineers and reduce the product test development time.

Early identification of test requirements was given high priority because of the need to identify any potential hardware additions, such as the analog self-test multiplexer, which might be required for manufacturing test. This early identification allowed these additions to be incorporated into the layout of the printed circuit board during the initial design phase.

We decided early in the design phase that the self-test multiplexer would need independent measurement capability so that it would not have to rely on other internal circuits for confirmation of functionality, and that it would be able to interface with the internal ADC for the built-in self-tests. These two paths for routing the multiplexer's output are necessary because there are two different problems to solve. First, at power on the self-tests are looking for process problems such as missing or backwards parts, and later, during

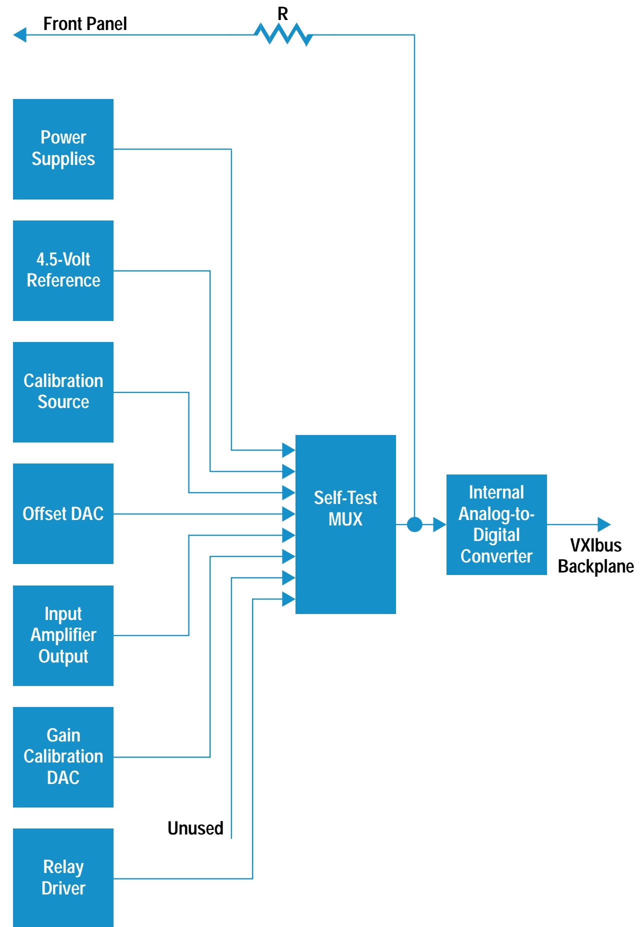


Fig. 3. Inputs to self-test multiplexer.

field verifications the self-tests are looking for component failures.

During surface mount production testing the accuracy of an external voltmeter is relied upon to verify important circuit parameters, while the internal ADC is used for built-in self-test measurements. Thus, from a design point of view it was necessary to determine which circuits would be critical to measure and whether the measurements could easily be done on the HP E1413 printed circuit board. The major implementation concerns were parts cost and printed circuit board area. The density of the HP E1413 does not allow much room for extra self-test parts. An eight-channel multiplexer was selected as a compromise between test coverage and printed circuit board area. Fig. 3 shows the major blocks that provide input to the HP E1413 self-test multiplexer.

The expanding kernel approach* has been used for structuring the tests throughout the development of all of the test suites. The first test is designed to measure several critical internal power supplies with one measurement. This "one measurement" constraint was mandated by the limited number of inputs to the self-test multiplexer.

Some of the power supply buses used in the HP E1413 are connected directly to the mainframe power supplies, while

* The expanding kernel test methodology selects a minimum set, or kernel, of circuitry that is assumed to be functional and then verifies the remaining circuits in sequence, building on the kernel.

others are derived from those supplies. It was decided to test only the four power supplies derived from the mainframe supplies. The test assumes that the five mainframe power supplies are functioning and so they are not tested. The hardware implementation consists of a wired-OR connection of the derived power supplies to one input of the self-test multiplexer.

The next major functional blocks tested with the self-test multiplexer are the outputs of the input amplifier, the reference voltage for the ADC, the calibration source, and the DACs used for calibrating the HP E1413.

The DACs for the calibration source, gain, and offset corrections are tested by setting the conditions of all zeros and all ones plus zero-ones and one-zeros combinations to test for stuck high or stuck low bits, while the input amplifier is tested for all input conditions, including overload (by using an external source).

The self-test MUX also verifies the relay drivers. The output of each relay driver is wire-ORed through a resistor to one of the MUX inputs. The drivers are activated one at a time, and since one side of the relay coil is hardwired to ground with the driver acting as an active pull-up, the relay driver is able to verify that the relay coil is activated.

The seven self-test multiplexer channels (the eighth is unused) in conjunction with the built-in self-test proved to be more than adequate to verify the performance of the HP E1413 before final calibration and test.

The final test of the HP E1413 is made up of two sections: calibration and analog verification. The calibration section performs a full-scale and tare offset calibration* for each of the 64 channels and the internal current source. It also verifies all of the internal calibration constants to ensure that they are close to their nominal values. Any deviations would indicate a possible fault.

Final Test Optimization

To meet the goal of a threefold improvement in analog verification times over other similar products, it was necessary to implement a set of stringent test selection criteria. Traditional test techniques usually extract only one bit of test data for each measurement. A traditional test suite was generated for the analog portion of this product and a list of these tests is shown in Table I along with their test times.

The test data yield can be increased by selecting tests that verify a number of different parameters at one time, or by selecting tests that verify a block of ports simultaneously. As the test data yield is increased, the length of the test suite will decrease. This occurs because more test data is extracted from each test, resulting in fewer tests being required to provide adequate test coverage. Also, since the number of executable tests is decreased, test times should decrease as well.

A good example of a high-data-yield test is the discrete Fourier transform (DFT). An analysis of the results of a DFT test can give information on noise, signal-to-noise ratio, aperture uncertainty, number of effective bits, harmonics and harmonic distortion (magnitude, frequency, and phase), integral and differential nonlinearity, missing codes, and spurious responses. While there are other methods of measuring these parameters, they would all require additional data processing and in some cases additional test system hardware. The DFT is able to measure all of these parameters with just one hardware configuration, a sine wave source and a low-pass filter, and one software analysis routine.

Table I
Traditional UUT Verification Test Suite

Test	Test Time (minutes)
Built-in test	3.0
Offset and noise (1 channel, 5 ADC ranges)	0.9
Offset and noise (64 channels, lowest ADC range)	1.2
Full-scale gain (1 channel, 5 ADC ranges)	1.3
Full-scale gain (64 channels, highest ADC range)	3.2
Integral linearity (5 ADC ranges)	5.0
ADC bandwidth (1V range)	0.6
Number of effective bits (DFT) (1V range)	0.9
Overload detection/recovery (1V range)	0.5
MUX switching speed (64 channels)— T_{on} and T_{off}	12.2
Total Test Time	28.8

The DFT test can also be used to verify the FET multiplexer of the UUT (unit under test), which would normally be verified by measuring such parameters as switching times and leakage currents for each of the 64 channels. While these parameters might be of some interest in themselves, the primary concern is whether the FET MUX is able to scan the input signals at the specified rates and accuracies.

If the main performance objective of the MUX is to maintain a specified measurement accuracy at the maximum scanning rate, then this is the manner in which it should be tested. In production testing the HP E1413, the 64 MUX channels (eight channels per SCP and eight SCPs per HP E1413) are connected in parallel to a sine wave signal source and a continuous scan is initiated that sequences through all of the channels at the maximum scanning rate. This results in a digitized waveform in which successive data points are measured on different channels. Any dynamic switching problems that are present will show up as an elevated noise floor of the DFT.

Even traditionally slow tests such as gain and offset can be redefined to improve their efficiency. Rather than measuring the full-scale gains of each channel one at a time (e.g., 100 readings at channel 0, 100 readings at channel 1, and so on), a scan list can be set up to scan all 64 channels sequentially (e.g., loop 100 times taking one reading at each channel during each iteration). This entails only one configuration for the UUT and the application of the same input to all 64

* A tare calibration measures and corrects for the customer's and the HP E1413's offset errors on each channel. Tare calibration is described in the article on page 25.

UUT Configuration \ System Setup	S1	S2	S3	. . .	Sn	
C1	Test 1	Test 3	Test 3 Test 5			
C2		Test 2				
C3			Test 4			
⋮						
Cn					Test n	

Fig. 4. System setups and UUT configurations are combined with their respective tests to indicate tests that share common data needs. In this figure tests 3 and 5 can both use the same data defined by system setup S3 and UUT configuration C1.

channels of the UUT. The data for all 64 channels is transferred to the computer in one large block rather than in 64 smaller blocks. This reduces much of the software overhead associated with computer I/O operations. The data for the 64 channels is sorted by channel and only the worst channel's results are reported to the user for pass/fail determination. Only in the event of a failure is all of the data returned to the user.

This scanning technique is used for almost all of the HP E1413 analog tests. The value of this approach is not just reduced test times. During the development phase of the product, several design and component problems were encountered that only manifested themselves during the scanning tests.

Data Duplication

Another area of opportunity for increasing test efficiency came from the recognition that many test suites contain duplicate measurement data. Since source setup and settling times are major contributors to test times, any duplication of measured data results in longer test times. For example, the data required for an integral linearity test includes plus or minus full-scale data, zero data, and other intermediate points. One data set can therefore be used for computing linearity as well as gain, offset, and noise.

Duplicate data sets can easily be seen by creating a table of system setups and UUT configurations as shown in Fig. 4. Grid locations that contain more than one test number indicate that duplicate data is being taken and are opportunities for test consolidation. This approach leads the test engineer to consider the test process from the point of view of data acquisition rather than test results.

Test Statistical Quality Control Process

Extensive use was made of audit and depend test types to provide additional data for statistical quality control (SQC) purposes and repair technicians. These tests were designed to verify the UUT's performance in the traditional manner, one channel at a time.

Audit tests are tests that are executed on samples of the production UUTs to ensure that the new test techniques, such as the scanned gain and linearity tests, are adequately verifying the UUT's analog performance. The test results are reviewed on a periodic basis to determine whether or not tests should be added to or removed from the audit list.

The tests classified as depend types are executed if a specified preceding test failed. The purpose of depend tests is to provide the repair technicians with additional data to aid them in locating the cause of a failure.

Fuzzy Logic Data Analysis

The HP E1413 project was also used as a test bed for a fuzzy logic tool that was developed as an alternative to statistical methods to aid in the analysis of prototype and production test data.[1] The fuzzy logic tool is especially useful for summarizing the test results from small quantities of products, even a single unit.

The fuzzy logic tool attempts to quantify the test results for a single UUT by emulating the analysis patterns of a human test engineer. The results of each test are graded by their proximity to nominal values and by their weight, or relative importance. The grading is accomplished by normalizing each test result relative to the test specification, determining its fuzzy equivalent, and then combining all of the results via a fuzzy rule base to yield a fuzzy conclusion and a fuzzy logic figure of merit, or fuzzy index. Each test can be weighted by applying a fuzzy logic hedge, which includes terms such as "very" or "slightly" that allow a specific test result to be emphasized or deemphasized relative to other tests. The ability to weigh the test results is of particular importance because it allows more emphasis to be placed on test results that are expected to be more accurate than others. Calibration point testing represents tests of this type.

The fuzzy index, which is much like a weighted average, indicates the overall performance of the UUT. This result can be used to compare multiple UUTs tested under the same conditions or to compare one UUT's performance over changing environmental conditions.

The fuzzy index has also proven to be a useful metric for tracking prototype development. The overall performance figure should show improvement relative to the product's target specifications as development progresses and circuit design is refined.

Signal Conditioning Plug-ons

The lack of any necessity for final assembly and the ability of the HP E1413 to compensate effectively for gain and offset errors within the SCPs enabled a test-it-once solution to be implemented during surface mount technology fabrication. The same testing techniques were implemented for the SCPs as are used for the HP E1413 motherboard.

Since the SCPs are designed to operate as part of the HP E1413 motherboard, either a known good HP E1413 is required as a test fixture, or a custom test fixture is necessary. The strategy chosen is to use a known good HP E1413 motherboard because it is the simplest and most cost-effective solution compared to a custom test fixture. It also allows a simpler maintenance and upgrade strategy.

To enhance the flexibility of the test process, the test is designed to allow operators to mix and match SCPs on the test motherboard. The software is designed to interrogate each of the eight plug-on locations to determine if an SCP is loaded and if so, its type, and then to load the appropriate test program.

Conclusion

The result of these efforts is a test process that minimizes throughput in production and at the same time ensures that all product specifications are met. All of the project goals were met, including those of no process defects at final test and the 3× reduction in test times as summarized in Table II.

Table II
Optimized Verification Test Suite

Test	Test Time (minutes)
Built-in test	3.0
Overload detection and recovery	0.5
MUX, ADC, bandwidth FFT (64 channels in parallel)	1.5
Integral linearity, gain, offset, noise (64 channels, 5 ADC ranges)	6.5
Total Test Time	11.5

Reference

1. B Kolts, "Fuzzy Logic Improves Small-Lot Data Analysis," *Evaluation Engineering,* Vol. 30, no. 5, May 1994, pp. 30-36.

Design Leverage and Partnering in the Design of a Pressure Scanning Analog-to-Digital Converter

The HP E1414 pressure scanning VXIbus analog-to-digital converter completes HP's VXIbus offering for jet engine and wind tunnel test applications by providing the ability to make pressure measurements.

by Richard E. Warren and Conrad R. Proft

Development of the HP E1414 pressure scanning VXIbus analog-to-digital converter (ADC) product was highly leveraged from the HP E1413 64-channel scanning ADC product. The development of the HP E1414 was also guided by a partnership with Pressure Systems Incorporated (PSI), a Virginia-based company. PSI is a company with expertise and high visibility in the field of pressure transducers and systems that calibrate and measure those transducers.

The HP E1413 scanning ADC was created to solve customers' needs for versatile and custom data acquisition solutions in jet engine and wind tunnel test applications. These applications require many channels of voltage, strain, resistance, and temperature measurements that must be acquired rapidly and accurately while the engine is operating under rigorously specified and controlled test conditions. These tests require very expensive test facilities. A key cost factor is how quickly the measurement system can make the necessary measurements and how flexible it is in being reconfigured for various test setups.

The HP E1413 performs these measurements with the needed flexibility. It offers high data throughput, scalable channel count (another card can be added to get more channels without speed degradation), mixed measurements (voltage, temperature, strain, resistance) in engineering units, and full data acquisition speeds (100,000 readings per second). Calibration is quickly and easily accomplished with an in-circuit calibration subsystem.

One measurement type the HP E1413 does not provide is pressure. This is especially critical in jet engine and wind tunnel test applications. These applications require many pressure channels in addition to the previously mentioned mixed measurements. Pressure measurements in these applications have the same requirements of high data throughput, scalable channel count, and results reported in engineering units at maximum data acquisition speeds.

With pressure measurements as the missing link, the project team searched for experts in the field of pressure measurements and set up a partnership. Pressure Systems Incorporated (PSI) is a recognized leader in electronic pressure scanning and has much experience in jet engine testing and wind tunnel measurements. PSI also has an existing product line of pressure measuring scanners and calibrators that provide all of the needed functionality to make the measurements. However, these scanners and calibrators do not operate within the C-size VXIbus platform that allows the high throughput and mixed measurements that customers often need.

Our systems approach to providing all of the desired throughput features and all of the required mixed measurements was based on the C-size VXIbus platform and the partnership with PSI. This approach combines the new HP E1413 and E1414 VXIbus modules with PSI pressure scanners and calibrators, other VXIbus modules, and compiled SCPI (Standard Commands for Programmable Instruments) programs. This combination provides the performance and measurement versatility required to satisfy these demanding applications.

Pressure Scanning ADC

The HP E1414 pressure scanning ADC is a C-size VXIbus module that allows the user to make scanned pressure measurements on up to 512 channels at rates up to 50 kHz when attached to pressure scanners manufactured by PSI (see Fig. 1). It provides measurements either in volts or converted to engineering units such as psi, Pascals, or mmHg. The measurements are made available to the VXIbus backplane at full scanning speeds and maintain accuracy to the specified 0.05% of full-scale pressure levels determined by the pressure scanners. The HP E1414 has most of the software features of the HP E1413 in addition to calibration modes specific to the pressure measurement environment. Fig. 2 shows the layout of the HP E1414 and E1413 VXIbus modules.

The majority of the hardware and software for the HP E1414 was developed first for the HP E1413 and directly reused with minimal changes for the HP E1414. The areas of hardware that are nearly identical include the VXIbus backplane interface circuits with register sets and address decoders, the CPU and engineering units conversion engine composed of the digital signal processing (DSP) chip and supporting memory parts, the trigger and timer circuitry, the FIFO memory and current value table memory control subsystem, the ADC, and the calibration subsystem. Much of the circuitry of these sections is implemented using field programmable gate

Fig. 1. The C-size VXIbus module containing the HP E1414 pressure scanning ADC.

arrays and programmable array logic (PAL). The PAL and all but one of the six gate array designs of the HP E1413 were reused without change in the HP E1414. Only the ADC gate array is different because of the special scanning measurements of the PSI scanners. However, even that redesign was accomplished by deleting from and adding to the HP E1413 design.

The physical printed circuit board layout of the HP E1414 was originally copied directly from the physical layout of the HP E1413 for those sections of the board that had little or no change. The areas of the board that had to be different between the two designs were simply added into the areas of the board layout where unneeded parts from the HP E1413A had been deleted. This greatly reduced the time and cost to produce the HP E1414 printed circuit board once the HP E1413 board was designed. Parts that were deleted from the HP E1413 for the HP E1414 include 60 of the 64 channels with the calibration relays for each channel and the signal conditioning plug-ons. This left ample space for the 50-watt switching power supply that supplies power to the pressure scanners, the controlled-rise-time address line drivers that select PSI pressure scanners, the extra RAM for storing real number coefficients and tables for 512 channels of pressure, and the HP-IB circuitry to communicate with the PSI 8400 mainframe and pressure calibration units.

The ADC calibration scheme of the HP E1413 was also used in the HP E1414. Even though the HP E1414 does not have all of the onboard calibration needs of the HP E1413 (the HP E1414 does not have resistance or current-source calibration needs), those calibration resources that are common were

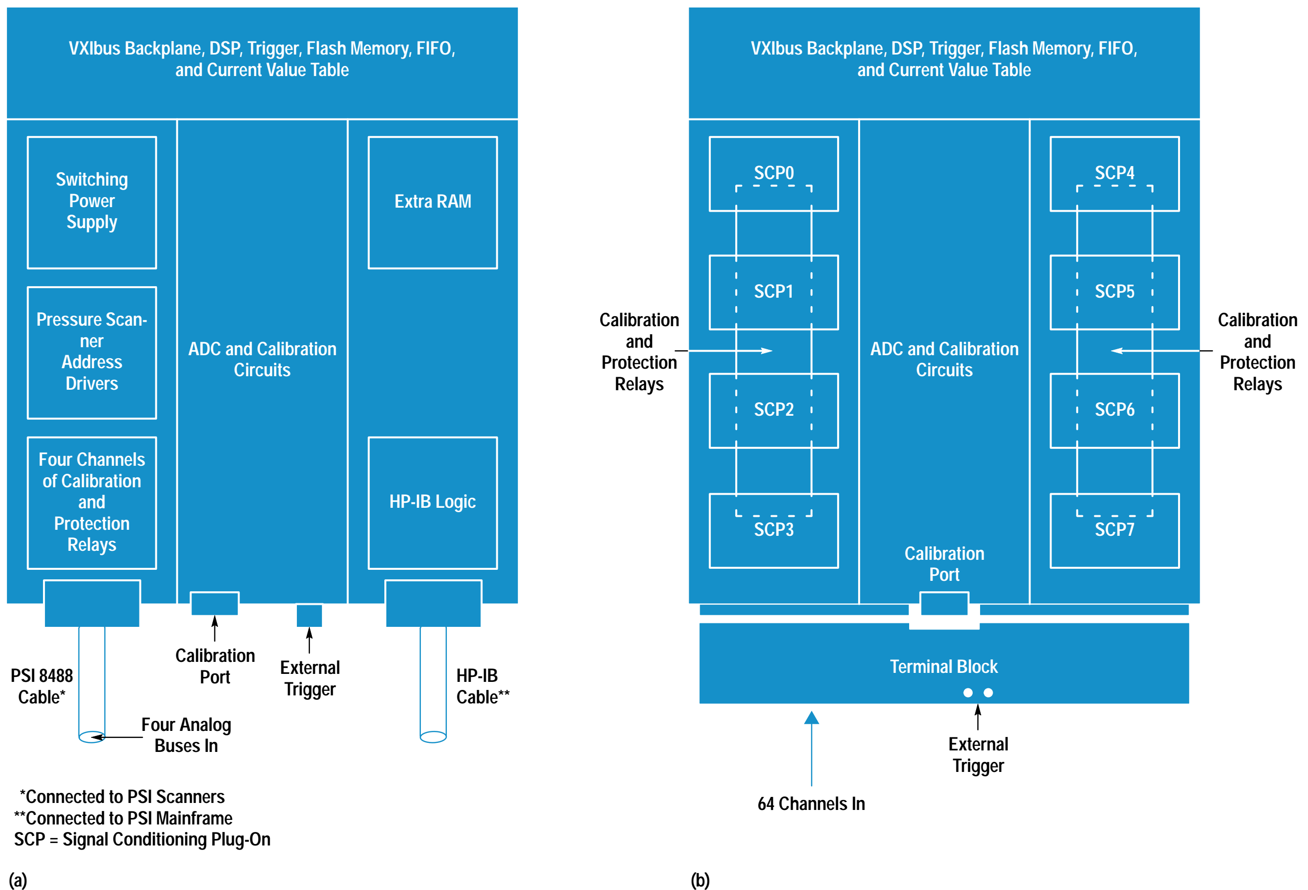


Fig. 2. The similarities and differences between the components on the (a) HP E1414 and (b) HP E1413 scanning analog-to-digital converters.

copied directly from the HP E1413 along with much of the software and calibration procedures for performing calibration of the HP E1414's ADC.

Software was leveraged from the HP E1413 in the areas of the status subsystem, trigger setup and control, accessing the FIFO memory and current value table, programming of the DSP flash memory, and the VMEbus memory subsystem. These areas represented over 50% of the software development.

Differences

The parts of the design where the two products differ greatly are actually fewer than the areas of commonality, but these differences are required to transform the HP E1413 into a pressure scanning ADC.

The HP E1414 has no signal conditioning plug-ons. It has only one intended input source, namely the pressure scanners from PSI. These scanners represent a family of products that all interface in a common way that is already preconditioned and eliminates the need for further signal conditioning of the input signals. Input signals are always preconditioned by PSI hardware to ±5V full scale.

The HP E1413 performs many different kinds of measurements depending upon the signal conditioning plug-ons, but the HP E1414 performs only voltage or pressure measurements. The HP E1413 provides built-in tables for converting measured volts into resistance, temperature, or strain. The HP E1414 performs a five-point pressure calibration and generates its own conversion tables to convert measured volts into pressure each time the pressure scanners are calibrated.

The HP E1413 can only scan across its onboard captive 64-channel multiplexer and make mixed measurements on those channels. The HP E1414 sources power to a maximum of 512 PSI pressure scanner channels and provides the necessary digital addressing to select and measure those channels. The current value table for both products can hold 512 readings, but the HP E1413 only makes use of the first 64. This is a good example of how the requirements for the HP E1414 were built into the HP E1413 design.

The HP E1413 can autorange the ADC to the best range for the level of the input received. The PSI scanner electronics always precondition the electrical output of the pressure transducer to produce a signal that is within a ±5V range for a full-scale input pressure. This means the HP E1414 only needs one measurement range, namely a 5V range. The HP E1413 uses binary ranging to aid in the conversion of volts to engineering units and at one time included ranges such as 1V, 4V, 16V, and so on. For the HP E1414 the 4V range was insufficient to cover the 5V output range and the 16V range would have resulted in reduced resolution. Fortunately, there was room for one more range-setting resistor in the precision resistor pack in the HP E1413. The 5V range resistor value needed for the HP E1414 was included in the empty resistor position and one precision resistor pack now serves the needs of both products.

The HP E1414 must interface to the PSI interface electronics and scanners in a predetermined way since these devices are a standard product from PSI and were in existence before the HP E1414 was conceived. The HP E1414 essentially replaces

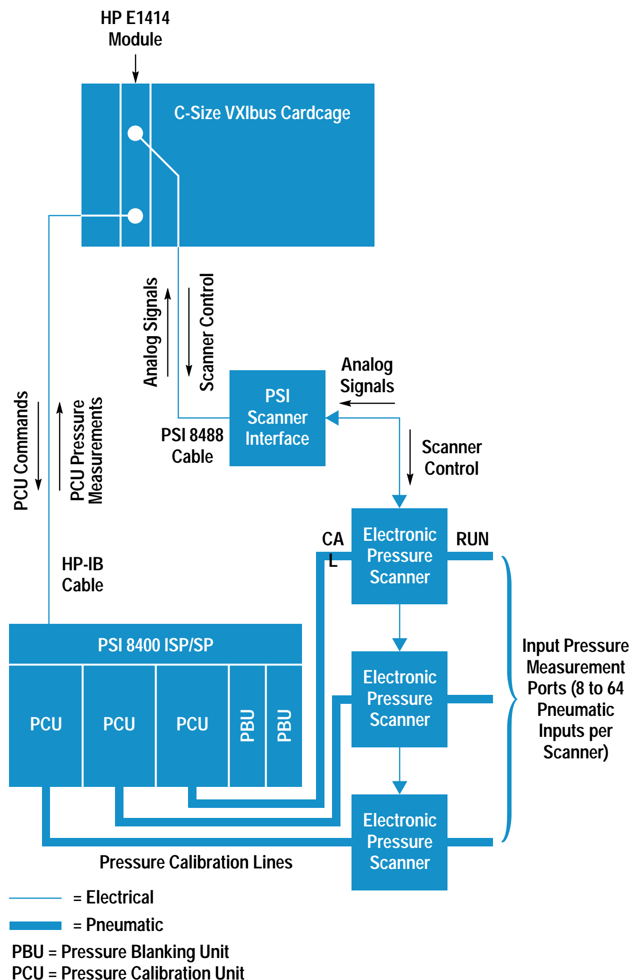


Fig. 3. Standard system configuration for the HP E1414 connected to the PSI 8400 for calibration and pressure measurement. Up to 512 pneumatic inputs can be connected to the HP 1414.

the scanner digitizer unit that normally plugs into the proprietary backplane of the PSI 8400 mainframe (see Fig. 3). The connection to the pressure scanners is via the PSI 8488 cable which includes four analog buses, digital addressing, and power to drive the scanners. (The PSI pressure scanner can permit up to four channels to be addressed at one time, each sourcing one of the four analog buses.) The HP E1414 contains all of the needed address line drivers as well as an onboard dc-to-dc converter power supply to provide up to 50W of power for the PSI electronics. This scheme greatly simplifies the use of the combined HP E1414 and PSI scanners by making all of the necessary connections with one common cable and no external power supplies required.

As shown in Fig. 2, the HP E1413 has channel multiplexing circuitry for 64 channels. It has a relay switch for each channel, which is used for calibration and input protection. Sixty of these channels along with their associated relay switches are deleted in the HP E1414 design. This leaves enough surface area to support the control and power circuits for driving the PSI scanner address lines and the HP-IB port. The HP E1414 itself only scans through four analog channels, representing the four analog buses in the PSI 8400 system.

PSI pressure scanners are limited to 50-μs minimum switching times between analog samples, which translates into 20-kHz maximum scanning rates. However, since up to four channels can drive their respective analog buses simultaneously, a technique called parallel address mode can be used, which effectively permits scanning the ADC at up to 80 kHz. This is accomplished by sampling one channel on an analog bus and then closing the next channel on that same analog bus immediately afterward. This permits each channel to settle for more than its required 50 μs before it is measured. However, since PSI 8488 cable lengths can range up to 100 meters and introduce capacitive loading, the scanning rates for parallel address mode are specified only to 50 kHz.

Pressure Sensor Calibration

The system-level calibration of the PSI scanners requires special software and hardware to guarantee that accurate pressure measurements are taken. This calibration begins by first positioning a piston contained in each electronic pressure scanner to the CAL position (see Fig. 3). The CAL position connects all input pressure ports on the pressure scanners to a single input called CAL. Then, five known pressures over the range of the transducer are applied to the CAL port via a PSI pressure calibration unit. The HP E1414 communicates via the HP-IB (IEEE 488.1, IEC 625) with the PSI 8400 mainframe, which houses the pressure calibration units. Up to 64 readings are taken and averaged by the HP E1414 with the analog signals obtained via the PSI 8488 cable. A five-point volts-versus-pressure curve for each scanner port is calculated, and a linearization table is generated and downloaded into the HP E1414's digital signal processor (DSP) memory for high-speed conversion. Once the calibration is completed, the piston in each scanner is driven in the opposite direction to the RUN position. In the RUN position unknown pressure inputs from the device under test are directed to the now calibrated pressure scanners. The pressure blanking units shown in Fig. 3 are used to fill empty slots in the PSI 8400 mainframe since its backplane consists of both electrical and pneumatic signals.

Different Memory Requirements

The HP E1413 has linearization tables already built into its memory, and it supports NIST-traceable transducers and custom linearization for up to 64 channels. The HP E1414 must support custom linearization tables and calibration coefficients for up to 512 channels, which requires much more RAM. This added memory is provided by increasing the available RAM from 128K 16-bit words to 512K words. The added memory requirement was anticipated early in the design of the products and decoding for it was built into the gate arrays and PAL of the HP E1413 so that these designs could be directly reused.

Common Development Tools

The same development tools were used for both products. The use of HP 9000 Series 700 workstations networked together within the design and manufacturing groups greatly enhanced the design leverage efforts by allowing easy access to common files and application software tools. Two of these applications, which run on the HP 9000 Series 700 workstations, are the Design Capture System (DCS) and the Printed Circuit Design System (PCDS). DCS allowed copies of design schematics to be easily copied and modified for the definition and production documentation of the board's circuit design as well as the internal circuit design of the gate arrays. PCDS allowed copies of one printed circuit board design to be easily copied and modified for the definition and production documentation of the physical printed circuit design. Deletions from the HP E1413 board design left about 50% of the board layout available, allowing the HP E1414-specific circuit sections to be placed in the spaces vacated by the deletions.

The RMB-UX programming language, a version of HP BASIC, allowed easy programmatic testing of the product features and performance specifications through computer-controlled test suites. Porting the HP E1414 driver to multiple platforms required both compiled SCPI and C programming, but the HP E1414 is easily programmed from the HP E1405/6 VXIbus command module[1] using RMB-UX over the HP-IB. This makes production testing, troubleshooting, HP ITG (Interactive Test Generator) programming, and operational testing much more accessible to a wide range of individuals who may not have an embedded computer or who are not familiar with programming in C. Since PSI was familiar with RMB-UX and the PSI 8400 is an HP-IB device, they found it quite easy to learn the HP E1414. RMB-UX also provided a simple means of updating the HP E1414's onboard flash program memory as firmware revisions were made. Test programs written in RMB-UX for exercising the HP E1413 circuitry were easily modified to provide similar testing of the slightly different HP E1414 circuit sections.

Common Schedules

The HP E1413 and E1414 were envisioned as two needed parts of a total solution to customers' problems and therefore they were both under development at the same time. The HP E1413 is the more generic and higher-volume product. With this basic project priority, the HP E1414 was always planned to be behind in the schedule by about two months. This allowed the HP E1414 to take advantage of the work done on the HP E1413 and avoid repeating some of the short-term design iterations that the HP E1413 experienced. Overall, the HP E1414 design was much less iterative than the HP E1413 because of the planned schedule spacing between the two products. This allowed the project team to work heavily on the specific HP E1414 designs while the HP E1413 team was deeply involved in solving problems that were common to both products.

Partnership between HP and PSI

PSI is highly experienced in solving real-world pressure measuring applications. The company was organized originally to take advantage of the development of piezoresistive pressure sensors when these devices became a reality. Their inputs on how to implement and verify calibrations and linearizations were invaluable in achieving the overall system accuracy that the HP E1414 is able to deliver when used with PSI's scanners and calibrators.

PSI has developed a very good customer base and an understanding of the nature of pressure measurement problems over many years. This knowledge allows the HP E1414 to be presented effectively to people who are interested in making these multichannel scanned pressure measurements. These customers may not previously have been users of VXIbus products, which offer broader capability. Conversely, many

of the users of VXIbus systems have not been able to take advantage of the PSI pressure measurement capability because PSI did not have an offering in the VXIbus format. Therefore, the HP E1414 allows PSI users access to a new architecture that is becoming more widely used.

The pressure scanners, the electronic interface hardware to the pressure scanners, and the pneumatic calibrators are all manufactured and sold separately by PSI. HP does not manufacture any of these items for use with the HP E1414, but without them the HP E1414 could not perform any pressure measurements at all. The HP E1414 has the built-in intelligence to automate the control and sequencing of the PSI hardware elements to provide a VXIbus pressure measuring system that takes advantage of the existing PSI hardware functionality.

The HP E1414 had to interface with an existing product definition and user model. To create a complete pressure measuring system solution from hardware that comes from two different companies, the HP E1414 had to be compatible with the operational paradigm of the already existing PSI 8400 system. This means that the pressure-measuring control language, terminology, channel-numbering scheme, and calibration methodology had to keep the same look and feel as in the PSI 8400 system. PSI channel numbers correspond to a numbering scheme of rack or modular pressure scanners. This channel numbering scheme identifies a particular address to send out on the PSI 8488 cable, and it also dictates which analog bus within the cable is sampled. It was also important that the HP E1414 take advantage of all of the work that has been done on SCPI to make system programming easier and faster. This presented the software designer with a challenge in merging the strengths of SCPI with the need to present a familiar interface to the experienced PSI equipment user.

The new HP E1414 pressure measurement language expands in some areas from the PSI language. For example, the command to configure the pressure calibration unit in PSI 8400 language is:

```
PC1 CRS, LRN, Pressure Mode, Tolerance, Maximum Pressure
```

where CRS refers to a particular PSI 8400 frame and slot where a pressure calibration unit might be addressed, and LRN is a number between 1 and 12 representing a symbolic name assigned to that pressure calibration unit. The symbolic name is then required for other commands in the PSI system. The corresponding HP E1414 SCPI language representation is:

```
CALibrate[:PRESsure]:MODE CRS, Pressure Mode
CALibrate[:PRESsure]:TOLerance CRS, Tolerance
CALibrate[:PRESsure]:MAXPressure CRS, Maximum Pressure
```

Note the adherence to the CRS designation. The LRN symbolic name is now an internally generated parameter, so the programmer only needs to remember the slot position of the pressure calibration unit in the PSI 8400 mainframe.

Another expansion of the SCPI language permits the programming and query of a specific parameter that is not available in the PSI system. This becomes very important when supporting the HP E1414 product from menu-driven software packages such as HP VEE[2] and HP ITG.

Testing the HP E1414/PSI System

Pressure calibrators, transducers, and the associated peripheral components can be very expensive since these are pneumatic devices that require precision mechanical and electrical fabrication. Fig. 3 represents a single HP E1414 configuration with PSI components. Some systems could be much larger and include five or more pressure calibration units, 500 to 750 pressure channels (which requires multiple HP E1414s), and multiple racks to hold the pressure scanners. Testing the software and hardware on such large systems required sharing expensive equipment between HP and PSI. The project team was able to develop the HP E1414 and its software driver on a system with just two pressure calibration units and only six pressure scanners. Multiple HP E1414s could have been used to simulate large-point-count systems under certain conditions. However, for the most part the project team had to rely on PSI for large-system testing, since they are the pressure transducer manufacturer and frequently assemble large-point-count systems to be shipped to customers.

To minimize the need for such a wide array of expensive equipment and to reduce the many-hour testing time required to verify proper operation of software and hardware, an HP 9000 Series 300 computer with a B-size VXIbus mainframe containing DAC cards was constructed by the project team to simulate the operation of the PSI 8400, calibrators, and pressure scanners. Each B-size DAC channel was connected to one of the four analog buses on each HP E1414. Some were chained from HP E1414 to HP E1414, and some had separate DACs. A DAC connected to an analog bus simulated a pressure calibration unit. Any HP E1414 channel measured on that analog bus read the voltage output from the associated DAC.

The simulator program was written in HP RMB and used the HP 98624A HP-IB card configured as a nonsystem controller. Another HP 98624A card was used to communicate with the B-size VXIbus frame. The simulator was written to simulate up to eight pressure calibration units and permit simulation of an indefinite number of pressure scanning ports when HP E1414s have their analog buses chained to the same DAC. The time investment in this simulator had big returns. Tests could run quicker since no real pneumatic devices were resident in the system. This approach reduced the wear on such devices when executing long comprehensive tests by eliminating them from the tests. Other engineers writing drivers for HP ITG could use the simulator rather than an actual HP/PSI system to do their development. This permitted concurrent development of the HP E1414 and HP ITG drivers. DACs were easier to manipulate under software control to simulate air leaks, failure conditions, and input voltage overloads. It was also easier to characterize the measurement integrity of the system by having direct control of input voltages.

Many of the regression tests for the HP E1413 were easily modified to test the HP E1414. Changes consisted mainly of channel-numbering differences between the two products. These tests included triggering, measurements, FIFO memory and current value table integrity, and interrupts. The self-test code for the HP E1414 was literally duplicated from

the HP E1413 code with only deletions and changes in channel count and numbering. This code initially took weeks to write for the HP E1413, but it was leveraged to the HP E1414 and tested in less than a week.

PSI currently runs every system they build, whether VXIbus-based or not, through the HP E1414 configuration to verify that various combinations of pressure calibration units and pressure scanners properly operate with the HP E1414. PSI's dedication to detail and understanding of pressure measurement problems has uncovered a variety of potential problems. Many of these types of problems would only be recognized by someone with many years of pressure scanning experience. Their experience saved many hours of trial and error that would have been wasted by the project team.

Customer Support and Training

Repairing or replacing an HP E1414 card when it fails is a straightforward operation that is well-defined within HP. The SCPI command *TST? performs a reasonably comprehensive self-test of the HP E1414's hardware and can pinpoint most failures. However, some problems may reside elsewhere such as in the computer platform, the PSI hardware, the tubing or cabling, or improper programming sequences. Determining where a problem lies can be frustrating when some or all of these factors combine.

Since the HP E1414 driver can run on HP 9000 Series 300 to 700 machines running the HP-UX* operating system, Radisys EPC-7 DOS/Lynx systems, or on the HP E1405/6 VXIbus command module, the combinations of possible failures escalate. In addition, PSI did not develop the HP E1414 driver nor do they have the resources to support all these platforms. To resolve these issues, a common troubleshooting technique and hardware configuration was established that is usable by both HP and PSI that has the best interests of the customer in mind.

The HP E1405/6 command module has already been the HP customer engineer's tool for isolating problems in the VXIbus environment. It is a helpful tool since it contains or can be programmed with all the HP VXIbus card drivers, is an HP-IB device and a VXIbus resource manager, has a terminal interface port, and can run HP Instrument BASIC for easy access to instrument drivers and hardware. In addition, the HP E1405/6 has a comprehensive debugging language to help resolve potential VXIbus card conflicts and failures. With HP Instrument BASIC and an external HP-IB flexible disk drive, the HP customer engineer can run diagnostic programs that can help isolate where problems might reside. This eliminates the need for the HP customer engineer or the PSI support engineer to be an expert with every computer platform.

PSI has written a comprehensive autoaccuracy and test diagnostic that communicates over the HP-IB with the HP E1405/6 command module from a portable PC or HP 9000 Series 300 computer. With this diagnostic, the HP customer engineer or the PSI support engineer can more quickly isolate system difficulties. If the conclusion is that the problem is related to the computer platform or the VXIbus hardware, then the HP support organization can help resolve the problem. If it is related to the pressure scanning equipment, PSI can help resolve the problem. In either case, a common diagnostic tool is available for both companies.

With this common diagnostic tool available, support training now focuses on understanding the system from that perspective. This tool helps reduce the number of variables resulting from multiple computer platforms.

Conclusion

Partnering with experts increases the certainty that the needs of the customer will be met by the product. The expert partner not only provides knowledge of the customer needs but also has gained valuable experience through time spent in actually solving those needs and understanding real-world applications. PSI had this type of knowledge in the area of pressure measurement in jet engine testing and wind tunnel applications, while HP provided the high-speed ADC conversion and at-speed engineering units conversion and VXIbus system knowledge to address the overall test system throughput issues.

Leveraging one product design from another can be very positive. The same team and tools that develop the first product can help develop the second. This second development effort can be done either simultaneously or in series, but the effort to do the second one is going to be less because all of the various activities involved in the development are now simply refinements or modifications of the original. The mental images of how the first product was done guide the developers in identifying how the required differences in functions can easily be achieved. This carries over into all aspects of the development including design, test, production systems, and learning products development.

The results we were able to achieve on the HP E1414 leveraged development have met nearly all of the objectives and schedules that we originally set forth. The HP E1414 was released to production with only two printed circuit board design cycles: the first prototype and the released version. This is always a project objective but in many cases unforeseen problems that come up late in a development can force yet one more turn of the board design. Through the work done on the HP E1413 and a two-month delayed schedule plan, the HP E1414 was able to incorporate many of the early changes from the HP E1413's first boards into the first board of the HP E1414. The overall system performance has met or exceeded all of our expectations and the combined PSI hardware (interfaces, scanners, pressure calibration units) and HP E1414 work well together. The system accuracies are as good as a PSI 8400 system can supply in nonVXIbus systems. Testing at both HP and PSI has shown that large-point-count systems can be configured and operated effectively while achieving 0.05% of full-scale accuracies.

Acknowledgments

The success of the HP E1414A design, implementation, testing, and release can be attributed to the HP E1413A project team: Von Campbell (project leader), Ron Riedel (ADC design), Gerry Raak (calibration circuitry and test software), John da Cunha (SCP design), Greg Hill and Dan Yee (VXIbus interface, FIFO and current value table, and trigger subsystem), Chris Kelley and Dave Rick (DSP firmware), and

Paul Sherwood (HP E1413A driver). Many considerations were included in their designs that made the port to the HP E1414A not only possible but also fast. We would also like to thank Pete Stone who was loaned to the project team to help port the HP E1413A and HP E1414A driver and test code to DOS and to the Lynx and HP-UX operating systems.

Special thanks go to Harry Trembley and Dan Ridenour from Pressure Systems Incorporated (PSI) who spent many hours helping us to understand pressure measurements, providing technical support, and supplying the necessary PSI pressure scanning equipment for the development of the HP E1414A pressure scanning ADC.

References

1. L.A. DesJardin, "VXIbus: A Standard for Test and Measurement System Architecture," *Hewlett-Packard Journal,* Vol. 43, no. 2, April 1992, pp. 6-14.

2. D.C. Beethe and W. L. Hunt, "A Visual Engineering Environment for Test Software Development," *Hewlett-Packard Journal,* Vol. 43, no. 5, October 1992, pp. 72-77.

HP-UX is based on and is compatible with Novell's UNIX® operating system. It also complies with X/Open's™ XPG4, POSIX 1003.1, 1003.2, FIPS 151-1, and SVID2 interface specifications.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

X/Open is a trademark of X/Open Company Limited in the UK and other countries.

Integrated Pin Electronics for Automatic Test Equipment

A single integrated circuit provides complete pin electronics for the HP 9493 mixed signal LSI test system. It contains a high-speed digital driver, an active load, a window comparator, and a parametric tester for setting a voltage and measuring current.

by James W. Grace, David M. DiPietro, Akito Kishida, and Kenji Kinsho

As system designers increase the degree of integration of their systems, the functionality of large-scale integrated (LSI) circuits becomes more complex and varied. As a result, to verify the operation of an LSI chip on a wafer and to certify a finished device as a good one, two major requirements must be met. First, the LSI test system must be equipped with higher pin count capability to test these highly integrated devices. However, in meeting this first requirement, the size of the system may increase greatly and inhibit the use of peripheral equipment in confined spaces. Second, the test system must have much higher throughput to minimize the test cost, which must be included in the selling cost of the device. To meet this requirement, the test system itself needs improved functionality. One way to provide this is to equip the system with many more resources. However, this may also expand the size of the system.

The HP 9493 mixed signal LSI test system (Fig. 1) is designed to meet both of these requirements while retaining the ability to operate in confined spaces. The number of digital pins is increased to 256 pins, double the number of pins of previous products, while peripheral equipment can easily be used close to the test head, which is the same size as in previous designs.

Architecture

The models of the HP 9490 Series employ the concept of having in one tester all of the resources necessary to provide the functions of digital, analog, time, and dc measurements. This allows many kinds of test signals, especially digital signals such as address, data, clock, and control signals, to be generated or received at all pins at the same time.

Fig. 2 is a block diagram of the architecture of the HP 9493 LSI test system. The digital test subsystem consists of the sequencers, capture memory, timing vector generators, per-pin digital electronics, and per-pin dc electronics. The synchro pipe consists of special hardware to synchronize operation of the digital test subsystem and the analog test subsystems such as the arbitrary waveform generators and the waveform digitizers. The timing vector generator and the per-pin digital electronics are the per-pin digital test resources. Up to 256 digital channels can be installed.

The timing vector generators, which generate the programmed test patterns, are located in the test system cabinet. There is one timing vector generator for each pin. The same number of digital and dc pin electronics blocks, located in the test head, work independently in accordance with the

Fig. 1. The HP 9493 mixed signal LSI test system has up to 256 pins and dual slim-design test heads to allow use of peripheral equipment in confined spaces. The system shown has two test heads, three monitors, and 2½ equipment bays.

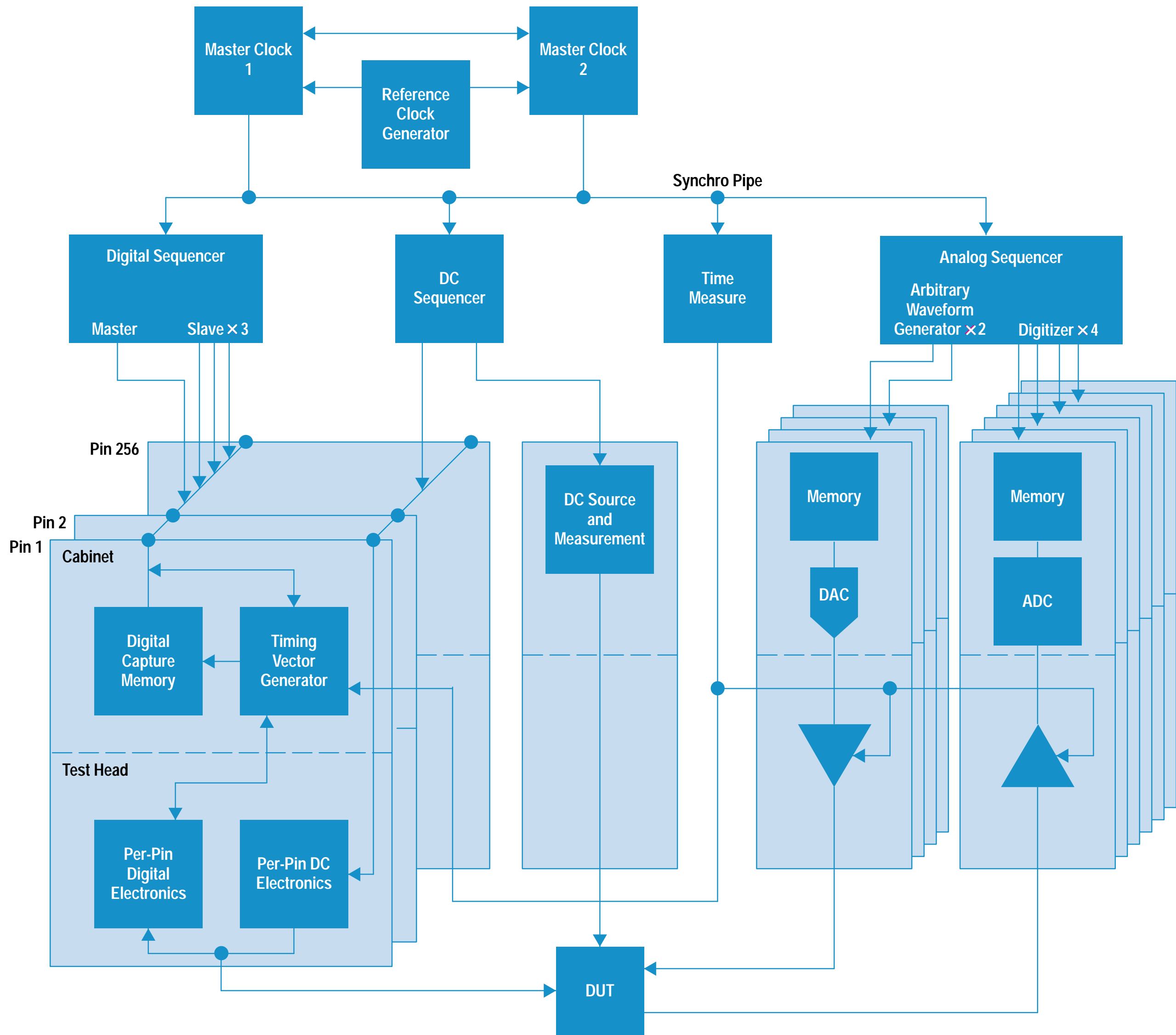


Fig. 2. Architecture of the HP 9493 mixed signal LSI test system. Each pin is equipped with both digital and dc resources.

defined patterns generated by the timing vector generators. The test head is the interface for test signals between the DUT (device under test) and the LSI test system. The pin electronics generate or receive digital signals and dc measurements at the front end of the DUT. This per-pin architecture gives the system improved throughput and the ability to interface various kinds of signals in the testing of LSI devices.

In addition to the pin electronics, many other hardware functions are built into the test head. The HP 9493 has a maximum capacity of 256 digital pins, and to reduce the size of the test system, the test head needs to be small. The test head was carefully designed to incorporate both small size and total functionality. This saves floor space and makes the system easier to handle.

Pin Electronics Functions

For LSI testing, the following functions are necessary:

- Driving formatted digital patterns to the DUT at arbitrary voltage levels
- Capturing digital patterns from the DUT and getting the results of timing measurements at arbitrary threshold levels
- Dynamically defining the load conditions at the output ports of the DUT
- VFIM (voltage force and current measure) and IFVM (current force and voltage measure), which are used for continuity tests, supply voltages, input currents, and leakage current measurements.

The pin electronics are responsible for performing these functions. The following circuits are necessary for providing various test capabilities in the HP 9493 mixed signal LSI test system.

- Pin driver. The pin driver generates the digital test signals to the DUT. The maximum data rate is 128 MHz.
- Pin comparator. The pin comparator receives the digital signals from the DUT and compares the signal levels with the reference voltage. The maximum data rate is 128 MHz.
- Active load. The active load receives and terminates the digital signals from the DUT. It also delivers the programmable current load to the DUT.
- VFIM. The VFIM function applies a constant voltage to the DUT and measures the current flowing into or out of the DUT.

- IFVM. The IFVM function applies a constant current to the DUT and measures the output voltage of the DUT. This function uses a successive approximation method and the programmable current of the active load.

We wanted to lower costs, increase functional pin capability, and double the pin count in the same test head volume. One way to achieve this is to combine functions within an IC chip and minimize the external components servicing the chip. Pin electronics have needed quite different technologies to perform their functions—one technology was used for the high-speed circuits (maximum frequency up to 128 MHz), while another was used for the precision dc circuits (resolution as high as 2.5 mV and 3 nA). We decided that the way to combine these functions was to design a custom monolithic analog ASIC (application-specific integrated circuit) that would include everything except the successive approximation circuits.

Pin Board on a Chip

This goal of combining the per-pin digital and dc electronics is realized in the PBOC, an acronym for pin board on a chip. The PBOC contains in a single chip a high-speed digital driver, an active load, a window comparator, a parametric tester for setting a voltage and measuring current, and control circuitry.

A functional block diagram of the PBOC is shown in Fig. 3.

The IC fabrication process was chosen for its device attributes of high speed and high voltage. Because of very low leakage requirements and variable chip temperatures, great effort went into minimizing all leakage paths, including device-to-device, device-to-substrate, and inversion paths. This was accomplished by using appropriate channel stops and metal guarding of the integrated components.

The objectives of the chip were:

- Low cost
- Eight channels on one board for a 256-pin system
- Minimum external components on the board
- Demanding linearity and accuracy specifications
- Compatibility with previous system designs.

Key features of the PBOC are:

- Pin driver, pin comparator, active load, and VFIM in one chip
- Operating frequency up to 128 MHz (64 MHz for large-amplitude pulses)
- Pin driver current to 35 mA
- Short-circuit protection
- Active load capability to sink or source 35 mA
- Two-nanosecond capture pulse width for the pin comparator
- Three VFIM ranges: 6 μA, 200 μA, and 6 mA
- On-chip measurement resistors
- On-chip range changing capability
- On-chip temperature compensation.

The main objective of the PBOC chip was to achieve a low-cost solution by including both analog and digital functions on one chip. The chip functions are linear within 0.05% to 0.1%, which simplifies calibration because it means that the functions only have to be calibrated at two points. This also lowers the cost of the system.

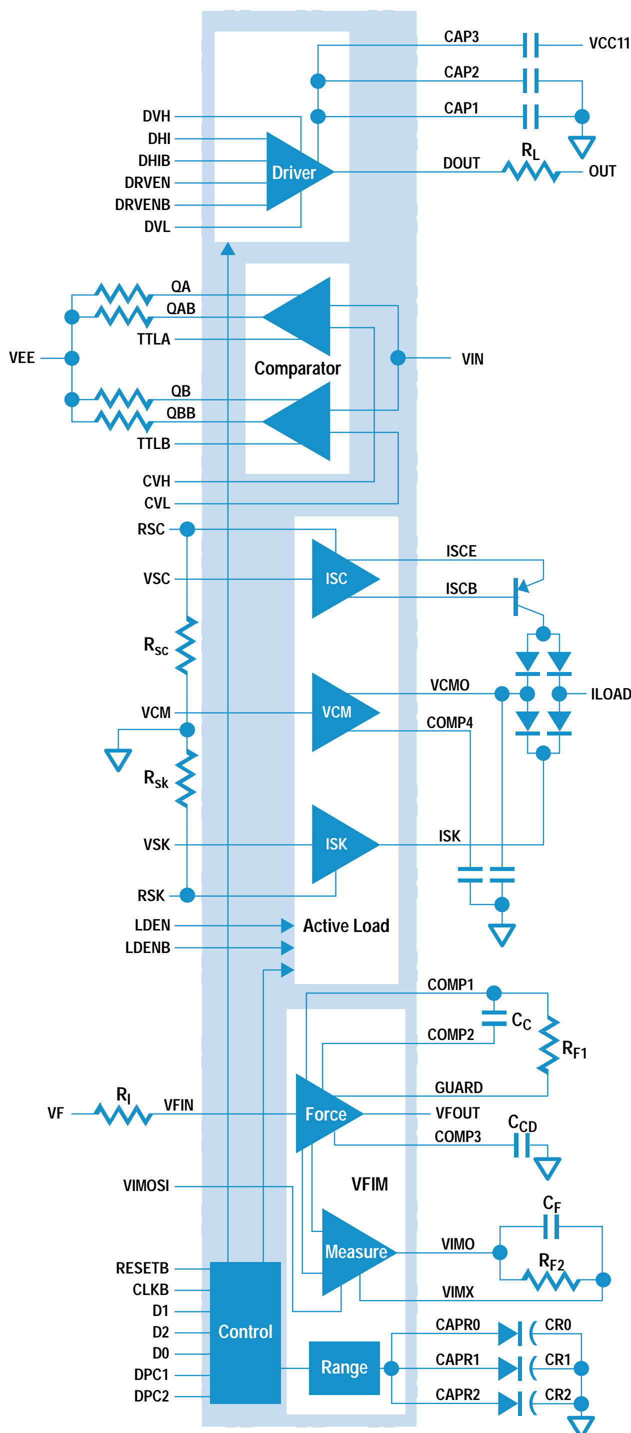


Fig. 3. Block diagram of the pin board on a chip (PBOC) integrated circuit.

Pin Driver. The pin driver drives the output terminal to either VOH or VOL according to the DHI and DHIB ECL inputs. A special override using TTL inputs sets the driver to either an on or an off condition by overriding the enable function. VOH and VOL are set by analog voltage inputs to DVH and DVL. Fig. 4 shows the functional block diagram of the pin driver.

The pin driver output can be put into a high-impedance mode by ECL inputs DRVEN and DRVENB in normal operation

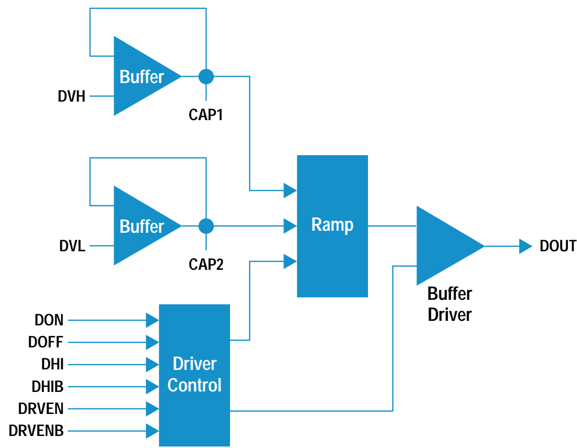


Fig. 4. Pin driver functional schematic diagram.

or by the internal DON and DOFF control functions, which are data inputs to the chip.

The pin driver provides a programmable output voltage of –2.1V to 7V with 12-bit resolution, 0.05% linearity, a 50-ohm output impedance, and a tristate output (high logic level, low logic level, high impedance). It can sink or source more than 70 mA for ac operation.

Pin Comparator. The pin comparator is a two-channel window comparator that compares the input signal against two reference voltages CVH and CVL. The ECL outputs (QA and QAB, QB and QBB) indicate whether the input voltage is higher or lower than each reference voltage. The outputs are balanced ECL10K-compatible signals driving 100-ohm twisted pair wires. TTLA and TTLB are TTL-compatible signals used for system setup calibration; their inclusion eliminates an external signal conversion circuit for the tester controller.

Fig. 5 is the functional block diagram of the pin comparator. The comparators have typical hysteresis voltages of 2.8 mV and typical offset voltages of 1 to 2 mV. The input range is from –2.1 volts to +7 volts. The comparator is capable of operating at over 128 MHz and has the ability to capture a pulse less than 2 nanoseconds in width. It has a combinational ECL logic output and an input impedance of 4 megohms.

Active Load. The active load loads the DUT with a specified current programmable from analog voltage inputs VSC and VSK. Fig. 6 is the functional block diagram of the active load.

If the voltage of the output terminal is higher than the commutation voltage (VCM), then the current programmed by VSK will be sunk from the DUT output terminal. If the voltage of

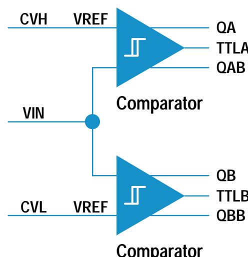


Fig. 5. Pin comparator functional schematic diagram.

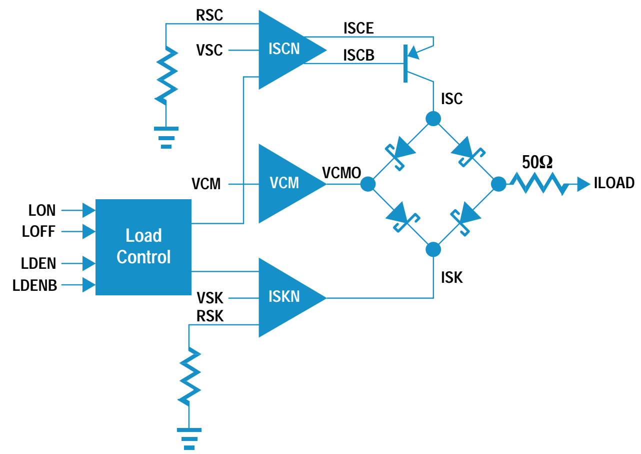


Fig. 6. Active load functional schematic diagram.

the output terminal is lower than VCM, then the current programmed by VSC will be sourced to the output terminal. When the voltage at the output terminal is near VCM, then the output current characteristics show a resistive impedance of 50 ohms. The voltage-current relationship is shown in Fig. 7. Notice that when the voltage (VCM ±DUT voltage) is greater than the sink current times 50 ohms or less than the source current times 50 ohms, the output becomes high-impedance and the DUT sees only the current being applied.

The relationship between the source current (ISC) or sink current (ISK) and the current control voltage (VSC and VSK) is given by the following:

$$ISC = VSC/RSC \times G.ISC$$

$$ISK = VSK/RSK \times G.ISK,$$

where RSC = RSK = 4 kilohms and G.ISC = G.ISK = 20 (typical).

The active load's high-impedance function is controlled by the enable signals LDEN and LDENB, which are balanced ECL10K input signals. In normal operation, a signal using TTL inputs can override the enable signals and activate or deactivate the load circuit. This condition is used for setup

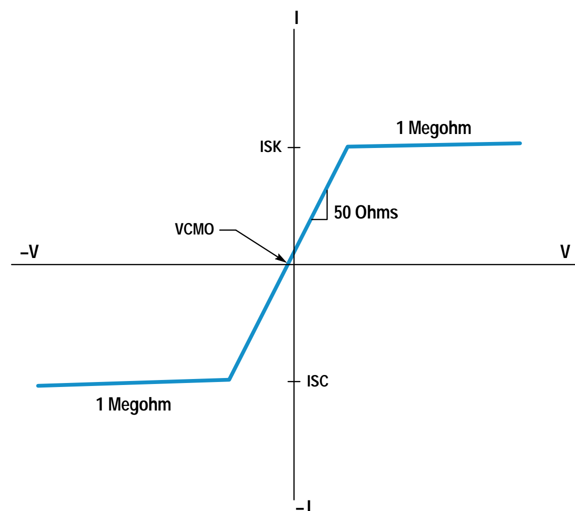


Fig. 7. Active load voltage-current relationship.

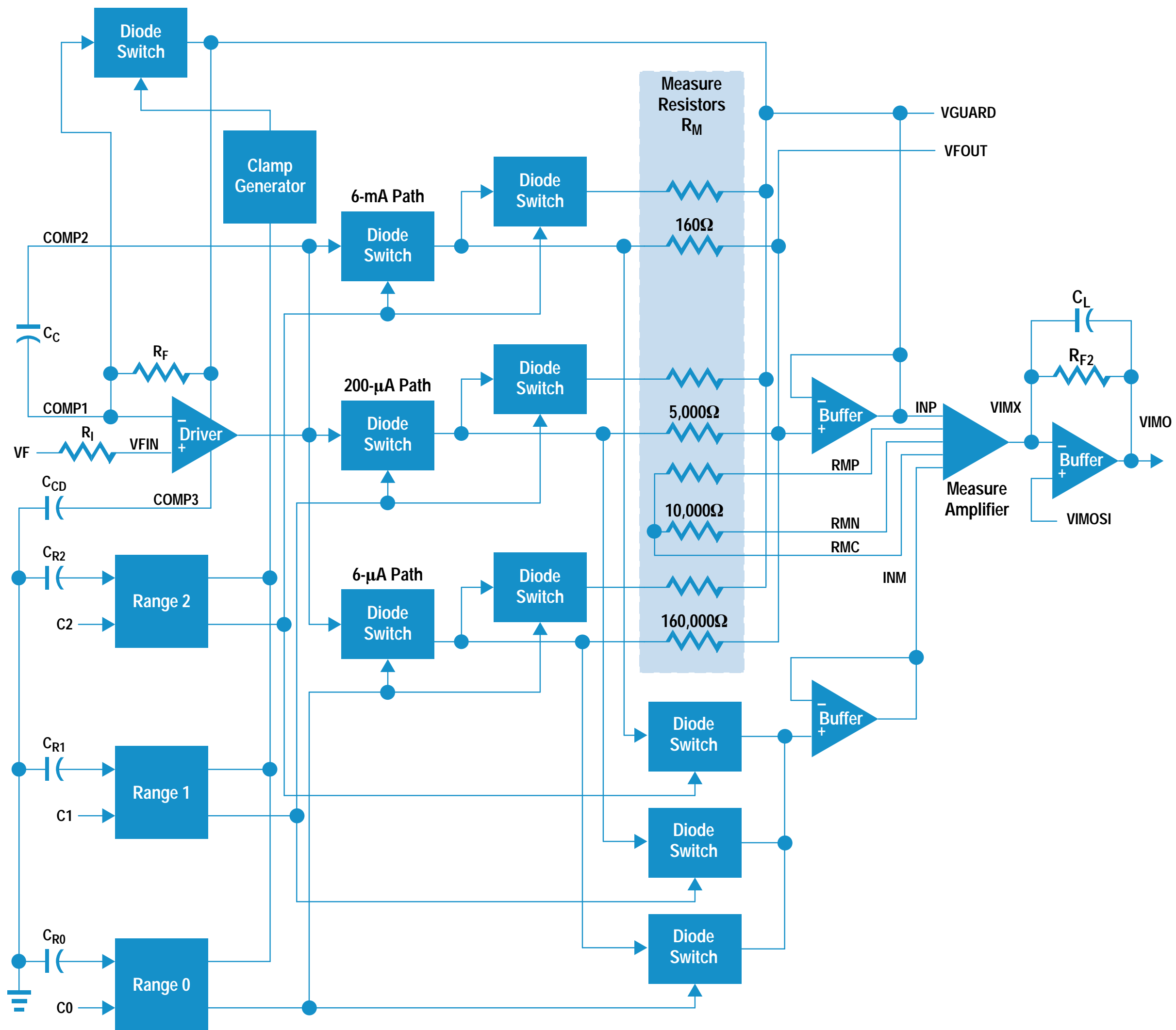


Fig. 8. Voltage force and current measure (VFIM) circuit functional schematic diagram.

and calibration. It is an abnormal operation in which excess power is dissipated on the chip. External to the chip, DRVEN and LDENB are connected together and DRVENB and LDEN are connected together such that only one circuit (driver or active load) is operational at any time.

In the off condition, the active load has a high-impedance output state and its power consumption is minimized.

The functions of the active load are realized with a few external components, including a p-n-p transistor, a Schottky diode bridge, two capacitors, and three resistors. The resistors are for setting source and sink currents and for adjusting the output impedance (50 ohms) in the resistive termination region.

The active load can sink or source from 10 µA to 35 mA with voltage compliance of −2.1V to 7V. It has 12-bit resolution, linearity of 0.1%, and an output impedance of 50 ohms. In the current sinking or sourcing mode, the output impedance is more than one megohm.

Voltage Force and Current Measure (VFIM) Circuit. The VFIM circuit is the portion of the PBOC that performs the precision voltage force, current sense, and current limiting functions of a voltage force driver measuring system. Design emphasis was on high resolution and accuracy spanning a range of force voltages from −2.1 to +7 volts and current sense ranges of 6 µA, 200 µA, and 6 mA.

Fig. 8 is the functional block diagram of the VFIM circuit. The VFIM input voltage VFIN, which is derived from the pin board circuitry, is an analog voltage source ranging from −2.1 to +7 volts. VFIN is buffered and applied to a current measure resistor R_M. A buffered loop compensates for the drop across R_M and produces VFOUT. The output current is measured by measuring the voltage drop across the terminals of R_M. Load current may flow in either direction through R_M. The resistor voltage drop is converted to a single-ended output, amplified to 4 volts full scale, and offset by +2.5 volts. The offset is needed to accommodate the operating voltage range of the pin board, which functions as both digital-to-analog and analog-to-digital converter. The result is VIMO, which ranges from −1.5 to +6.5 volts. A current limiter limits the output current to about 135% of each range.

The current measurement ranges are controlled by an internal register programmed by the test system controller. Three values of measure resistors are switched in and out of the circuit to achieve the three current ranges. The rate of transition between current measure ranges is controlled by external capacitors and the range blocks in Fig. 8. During range

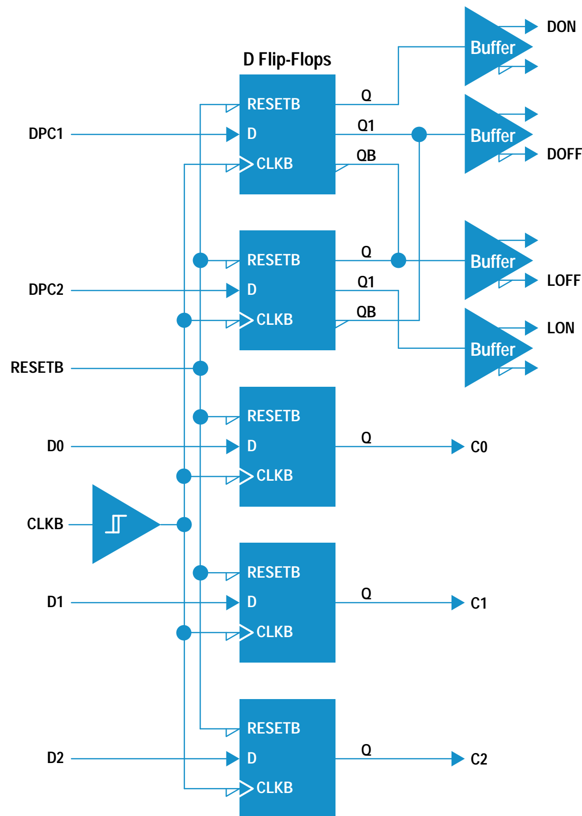


Fig. 9. Control circuit functional schematic diagram.

change transitions, the feedback resistor R_F is shunted by a low resistance, thereby decreasing the response time of the voltage force loop. This feature effectively reduces the height of the spike generated during range changes.

Nine external components are required for the VFIM function. Capacitor C_C is required for compensation and stability of the voltage force loop. Capacitor C_{CD} is required for stability of the drive amplifier. Capacitor C_L forms a low-pass filter in the current measure path. Resistor R_{F2} sets the gain of the current measure path. Resistor R_F is required for the voltage force feedback path and R_I is a matching resistor for the VFIN input. Capacitors C_{R0}, C_{R1} and C_{R2} are required to control the speed of the range changes.

The VFIM circuit can measure currents from 3 nA to 6 mA in three ranges of 6 μA, 200 μA, and 6 mA full scale. The circuit has voltage and current linearities of 0.05%, current common-mode rejection of 0.1%, and 12-bit voltage and current resolution.

Control. The control circuitry (see Fig. 9) controls the VFIM range changes and turns the driver and active load circuits on and off. The inputs CLKB, RESETB, DPC1, DPC2, D0, D1, and D2 are TTL signals. The data is captured in D flip-flops, which then logically control the functions of the chip.

The data input, reset, and clock signals are generated from other HP custom integrated circuits so as to minimize external components and reduce the pin count of the ICs used. This greatly reduces the pin board area needed for supporting eight channels.

Design Limitations and Challenges

The PBOC is used as a retrofit for an existing automatic test system. This constrained the design of the chip. It limited the choice of HP silicon fabrication processes and required the use of preselected automatic test system voltage supplies and conformance to preselected control conditions. It also required the use of diffusion resistors (nonlinear and temperature sensitive) and high-voltage biasing of components. Voltage breakdown limitations, leakage currents, and chip temperature excursions had to be dealt with.

Design challenges for the pin driver included a fast slew rate, accurate amplitude, a low-leakage tristate output, nonloading capacitance in the off state, an all n-p-n transistor push-pull design to minimize power consumption, and driver enable and disable capability.

The active load required conversion of input voltages to output currents, the ability to enable and disable the sink and source currents, the ability to monitor beta and compensate for base currents, and a linearized current range.

The comparator required fast capture times, small propagation delays, small input hysteresis voltage, and high input impedance.

The parametric tester required range changing without overshoot and within time limits, 3-nA sensitivity and accuracy maintained over chip temperature and voltage excursions, linearity of 0.05% for both voltage output and current measurement, and conversion of the current being measured to a voltage output.

Solutions and Implementation

The driver consists of buffer circuits for setting output levels and a ramp circuit to drive the output unity-gain buffer-driver. The buffer-driver consists of an all n-p-n transistor configuration. The push-pull effect is achieved by a variable pulldown current dependent on the state of the driver. A controlling circuit coordinates the ramp and push-pull capability of the buffer-driver.

The active load consists of transconductance amplifiers to change voltage inputs to known currents. These currents are then amplified with a gain of 20 to produce the ISC and ISK output currents. The circuit must attain its programmed output value from an off condition in 20 nanoseconds. To make this possible, the current amplifier capacitors are precharged so that the amplifiers reach the proper biasing voltages quickly.

The ISC output current needs to be corrected for the base current of the external p-n-p transistor. This is done by sensing the base current and reapplying it within the current amplifier. This corrects for gain variation, nonlinearity, and output impedance variation over the entire current range.

The ISK output current requires a similar correction for the internal n-p-n transistor. In addition, the current produced by the input transconductance amplifier is mirrored and then amplified 20 times. Additional correction is provided by a linearizing circuit. Again, the base current of the output transistor is sensed and reapplied within the current amplifier. This, as before, corrects for gain variation, nonlinearity, and output impedance variation over the entire current range.

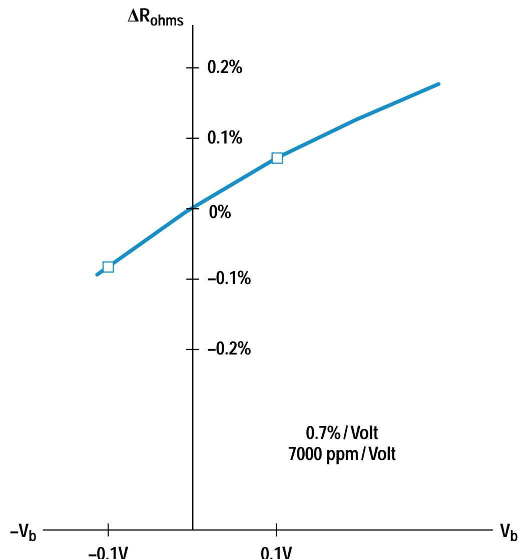


Fig. 10. Nonlinear diffused resistor curve, showing dependence of resistance on bias voltage.

The comparator requires input current cancellation to produce the high input impedance. This is accomplished by sensing a replication of the base current of the input transistor and subtracting it from the input node.

VFIM Current Measuring Resistors. One of the most creative solutions used in the PBOC was the design of the measure resistors R_M for measuring currents in the VFIM circuit. Diffused resistors are inherently nonlinear when used at different voltage bias conditions. Our objective was to produce resistors that have linearities better than 0.05% over their full operating current range, do not change value over the common-mode voltage range, and are not affected by leakage paths. The VFIM circuit measures currents from 3 nA to 6 mA and tolerates common-mode voltages from –3 volts to +8 volts about ground. For these ranges we needed to have 160-ohm, 5000-ohm, 160,000-ohm, and 10,000-ohm resistors that had matching characteristics over the operating current and voltage conditions and temperature variations.

Diffused resistor values vary with biasing potential (Fig. 10). By biasing a resistor so that one end is forward biased by a small amount and the other end is reverse biased the same amount with respect to the epitaxial island, the nonlinear effects are cancelled. The resulting resistor element has the same value whether it is operating at full current or zero current.

The solution is to bias a resistor element so that the potential at its center equals the resistor island potential as shown in Fig. 11. This makes the voltage excursion of the resistor ends about the center equal, so the resistor retains its value and linearity. The larger the number of elements used the more linear the resistor becomes. We settled upon five elements to produce each resistor. The resistors R_1, R_2, and R_3 are p-type diffused resistors in an n-type epitaxial island. The measurement resistor R_1 is symmetrically forward and reverse biased by no more than 0.1 volt. This minimizes the forward biased injection and the forward biased current is cancelled to a certain extent by the reverse leakage. Resistors R_2 and R_3 bias the island at midpoint. These resistors also bias the resistor island to the substrate potential to eliminate this leakage path. Fig. 11 shows the resistors and the resistor-to-island and substrate-to-island parasitic diodes.

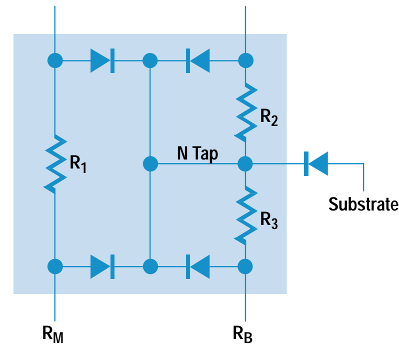


Fig. 11. One element of the measure resistor, showing parasitic diodes.

The way the resistors are biased also helps eliminate leakage currents so that even at high temperature, the 3-nA capability is kept intact. Fig. 12 shows the five elements that make up a measuring resistor. They form a biasing string and a complete measurement resistor. Low-impedance drivers drive the bias string of R_2 and R_3 resistors which bias the R_1 resistors at midpoint and remove leakages to the substrate.

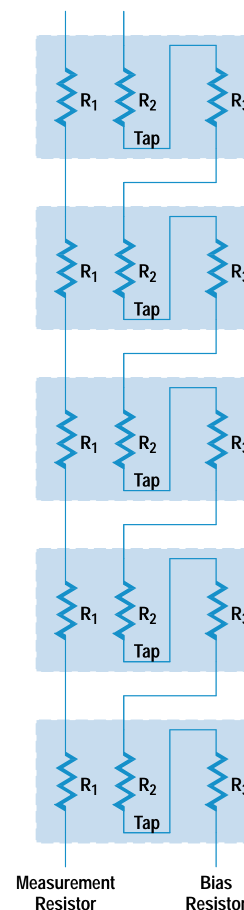


Fig. 12. The complete measure resistor consists of five elements in series. The R_2 and R_3 resistors form a bias string.

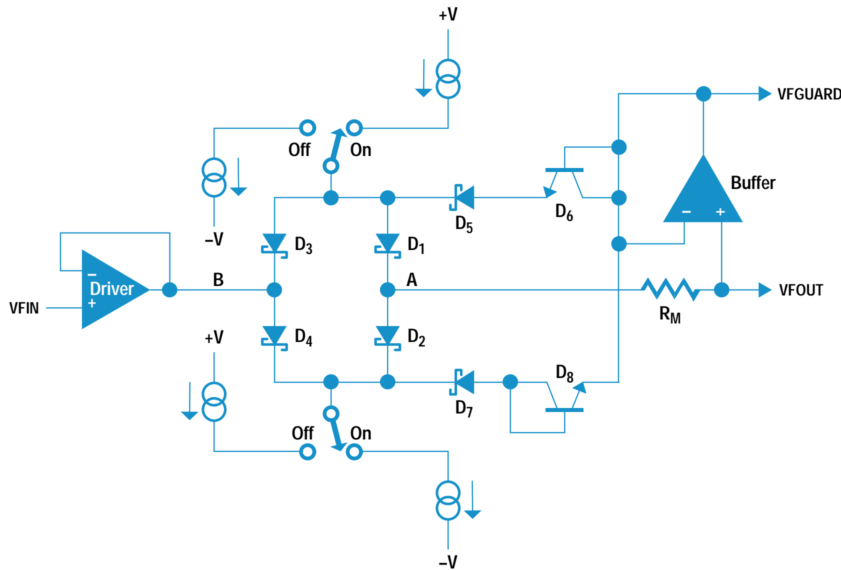


Fig. 13. Diode switch biasing.

The current measure amplifier of Fig. 8 has its resistors embedded in the measure resistor region so that mismatch and thermal effects are alleviated. These resistors also have five elements which are biased at their midpoints. This allows the measure amplifier output current at node VIMX to be an exact replica of the current being measured times some amplification factor determined by the resistor ratios. The output current sets up a voltage drop across an external precision resistor. The resultant voltage is measured by other analog-to-digital converters in the system.

The layout of the measure resistors ensures good thermal equalization between the elements. Good matching of resistors differing in value by 1000:1 was accomplished by series and parallel connection of resistor elements of the same width, as is customary in IC design. For example, 40 1280-ohm resistors are connected in parallel to make a 32-ohm resistor, while two 16-kilohm resistors are connected in series to make a 32-kilohm resistor. The 1280-ohm and 16-kilohm resistors are of the same type and width, differing only in length.

Leakage. Leakage is intolerable in a VFIM circuit of this type. Integrated Schottky diode bridge switches used for connecting the measurement resistors to other circuitry need to have very low leakage in the off state. When measuring current, two of the measurement resistors are connected to Schottky diode switches that are idle and reverse biased. The reverse-bias diode currents are large enough to affect the current measurements, especially at higher temperatures. In addition, the diode leakage current is a function of the reverse bias voltage. The matching of diode leakage is very good when bias conditions are equal.

Our approach was to back-bias the diodes by equal amounts so that leakage into a nodal point equals the leakage out of the node. The biasing scheme is shown in Fig. 13. When diodes D_1 and D_2 are reverse biased, they have equal bias potentials. This allows the current going into node A from diode 2 to equal the current out of node A to diode 1. The voltage needed for operation of the driver at node B is about 1 volt. Diodes D_5, D_6 and D_7, D_8 set up the equal biasing potential. Since no loading occurs from node A, VFOUT is not affected by these leakages.

Ranging Speed. To minimize or eliminate output voltage spikes during range changes and to minimize the range change time, the driver response speed needs to be increased during range changes. This is accomplished by shorting out the feedback resistor R_F during the range change period. A diode switch is used to short out the feedback resistor for the drive amplifier during the period of range change. Fig. 14 shows how the diode switch controls the voltage force driver. This allows the driver to react much faster by charging and discharging its compensation capacitor through the switch instead of through the resistor. The compensation capacitor C_C is quite large to provide high stabilization for all load conditions at VFOUT.

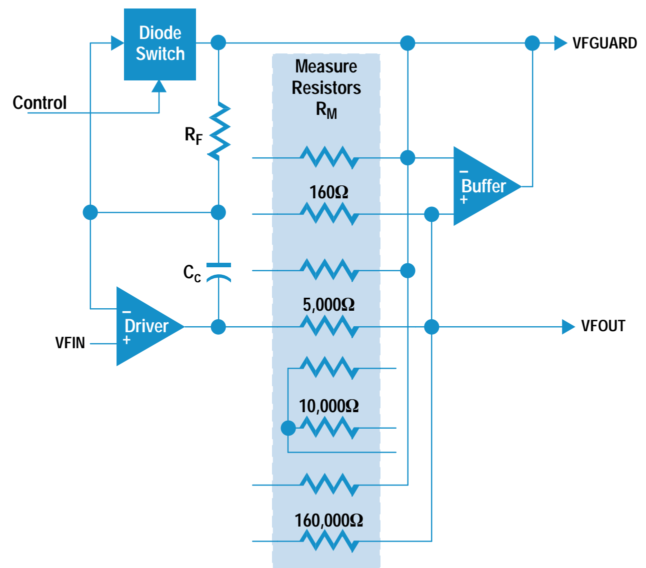


Fig. 14. Simplified schematic diagram showing the method of shorting the feedback resistor to improve driver response speed.

Conclusions

An integrated circuit was successfully designed and implemented to allow all of the pin electronics of the HP 9493 LSI test system to be included in one IC package with minimal external components. This makes it possible for a single board to hold eight channels, resulting in a compact test head size for a 256-pin system.

Acknowledgments

The following people have worked on this project in some capacity and the authors wish to acknowledge and thank them for their contributions: Huong Nguyen, Hiroshi Sakayori, Shinichi Tanida, Toshio Tamamura, Jimmy Chua, Kevin Martin, Chris Schiller, Bill Murphy, Dave Bigelow, Brian Levy, Pat Byrne, and Dan Hamling. We also wish to thank Jennifer DeNeve and Patti Friesen for the layout of the chip.

CMOS Programmable Delay Vernier

In the HP 9493 LSI test system, CMOS delay verniers replace the usual bipolar technology and are integrated with digital circuitry to produce a high-performance timing generator in a single monolithic CMOS VLSI formatter chip. This solution achieves bipolar-equivalent resolution, skew, and jitter performance with significantly lower power, cost, and circuit board space.

by Masaharu Goto, James O. Barnes, and Ronnie E. Owens

The HP 9493 is a mixed-signal LSI tester with a per-pin digital test resource architecture designed to offer the user test generation flexibility and ease of use. The timing vector generator is a key per-pin resource that generates and captures digital waveforms going to and coming from the device under test (DUT). Each DUT pin has an independent timing vector generator channel, allowing the user to select arbitrary timing, waveform format, and logic pattern without having to consider resource conflicts. Each timing vector generator consists of vector memory, formatter logic, and delay verniers, which are essentially high-precision programmable delay lines.

Traditionally, to meet the speed and timing performance requirements, high-precision delay verniers have been implemented with bipolar ECL technology.[1] However, these off-the-shelf delay verniers have the disadvantages of high cost and high power consumption. In addition, as separate packages, they increase board-level interconnections and package count. To address these problems in our system, the delay verniers were implemented with CMOS circuitry, allowing them to be integrated with the formatter logic into a single CMOS VLSI chip called ACCEL2 (Fig. 1). ACCEL2 was designed at the HP Integrated Circuits Business Division's Fort Collins Design Center and fabricated using HP's CMOS34 process.

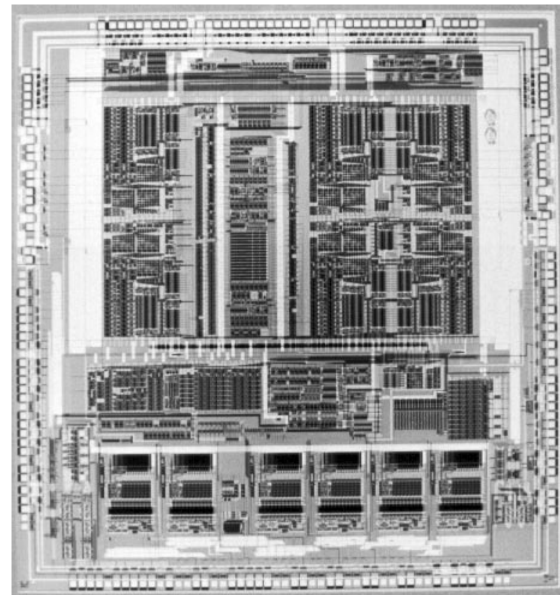


Fig. 1. ACCEL2 chip.

The benefits of this approach are evident, but there are many challenges in designing a CMOS timing system with the same level of performance as the bipolar counterpart. A stable, low-noise timing system is essential in a mixed-signal LSI tester. For example, testing state-of-the-art analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) requires jitter performance of 20 ps rms or less. CMOS gate propagation delays are very sensitive to temperature and supply voltage, and CMOS implementations of standard circuits such as operational amplifiers typically have higher noise than their bipolar versions. Any use of CMOS in timing system applications must circumvent these limitations.

In previous attempts to use CMOS in this setting, feedback techniques such as phase-locked loops have been used to stabilize delays. However, such an approach often compromises jitter to an extent that is unacceptable in a mixed-signal environment. For ACCEL2, we developed a method of stabilizing delays while maintaining jitter at levels close to the theoretical minimum of the CMOS FETs. Our approach employs custom CMOS design in the delay verniers and on-chip dynamic power compensation to minimize the temperature delay sensitivity. In addition, the problems of supply and temperature variation were addressed at the system level. The combination of these techniques resulted in a timing vector generator module with performance equal to or better than a system with bipolar delay verniers, but with substantially lower cost and power dissipation. In a high-pin-count VLSI tester, these savings can make an important contribution to the performance and value of the product.

Test System Block Diagram

A block diagram of the HP 9493 mixed signal LSI test system is shown in Fig. 2 on page 43. The digital test subsystem consists of the sequencers, capture memory, timing vector generators, per-pin digital electronics, and per-pin dc electronics. The synchro pipe consists of special hardware to synchronize operation of the digital test subsystem and the analog test subsystems such as the arbitrary waveform generator and the waveform digitizer. The timing vector generator and the per-pin digital electronics are the per-pin digital test resources. Up to 256 digital channels can be installed.

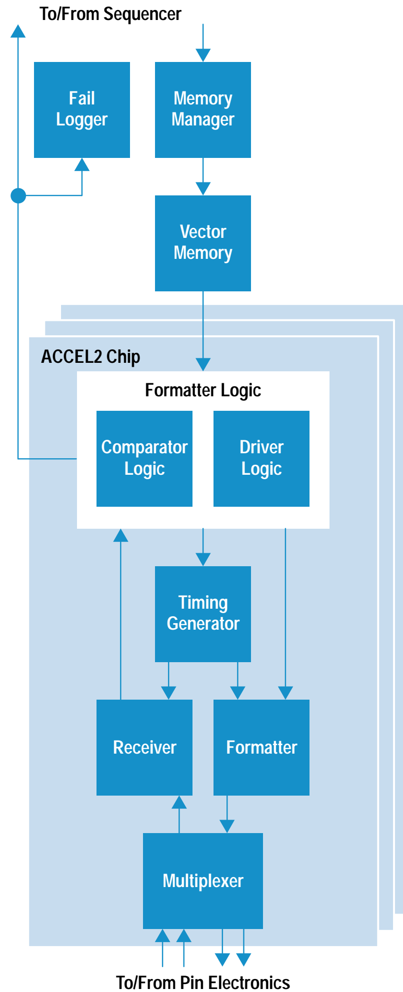


Fig. 2. Timing vector generator block diagram.

The timing vector generator block diagram is shown in Fig. 2. The memory manager, shared by eight timing vector generator channels, receives vector addresses from the sequencer and gives them to the vector memory. Test vectors stored in the vector memory are given to the formatter logic, which controls the formatter, receiver, and timing generators. A multiplexer is placed after the formatter and the receiver to support channel multiplexing and dual-test-head switching capability. The formatter logic, formatter, receiver, multiplexer, and timing generators are integrated in the ACCEL2 chip.

ACCEL2 Timing System

Fig. 3 is a simplified block diagram of the ACCEL2 timing generator. A coarse edge counter, implemented in standard CMOS digital logic, is used to generate a variable coarse delay ranging from 1 to 2^{14} periods of the system master clock. Because the counter terminal count output CE1 has significant jitter and skew, the edge is retimed by a flip-flop that we term the "last flip-flop." This block represents the final point along the path of the timing edge at which it has a fixed time relation to the system master clock. The last flip-flop is clocked by a clean version (MC) of the system master clock MCLK. After propagating the delay vernier, the delayed edge FE propagates to the drive or receive sections of the chip to control the timing of signals driven to the DUT or sampling of waveforms received from the DUT. There are six such timing generators on the ACCEL2 chip: four for the drive side and two for the receive side.

Fig. 3 illustrates a number of features that influenced the chip design. First, effective electrical isolation was needed between the standard digital part of the chip and the delay verniers and other timing-sensitive blocks such as the drive and receive circuitry. This required low-noise design techniques such as power supply isolation and control of rise and fall times in critical sections.

Second, a number of parasitic delays exist in any real implementation. Examples are the clock-to-output delay of the last flip-flop and the minimum delay through the time verniers. The sum of these delays makes up the intrinsic delay, which is specified in Table I. This delay must be stable.

Third, the design of the delay verniers is simplified if only one type of timing edge need be accurately delayed. In our case, it is the falling edge of the negative-going pulse from the coarse counter. As an illustration of this, the clock-to-output delay of the last flip-flop, which is a component of the intrinsic delay, is more easily controlled if only one transition is important. This is also true for the delay vernier itself.

Fourth, in this timing system, edges are separated by a variable time interval that depends on the setting of the coarse edge counter and the PCLK (period clock) frequency. This variation complicates the design of the delay verniers.

Fifth, since the coarse edge CE has a timing resolution of one master clock period, the delay range required of the delay vernier is related to this period. On ACCEL2, the required range is 16 ns.

Timing Specifications

The timing vector generator module must present highly accurate and precise waveforms to the device under test (DUT). This directly translates into stringent specifications for the on-chip timing system and delay verniers in particular. Table I gives the system requirements for the ACCEL2 timing system. These specifications are as stringent as those for the HP 9491A test system, which used external bipolar delay lines. Achievement of these specifications on a CMOS chip with a large amount of high-speed digital logic was the primary design challenge of the project.

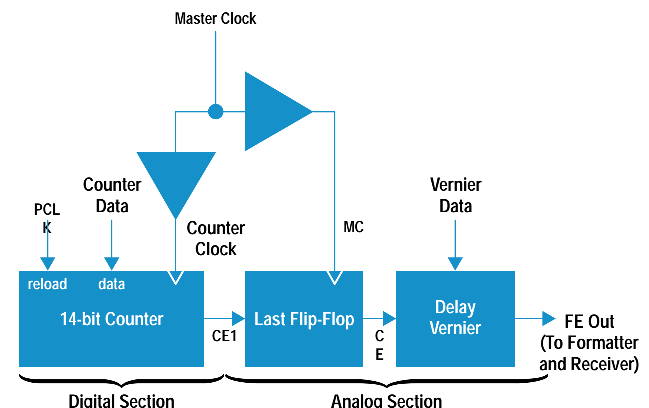


Fig. 3. Simplified block diagram of the ACCEL2 timing generator.

A few observations may be made regarding these specifications. The resolution shown is the user-programmable resolution, and linearity is given in terms of this step size. Because of the nature of the calibration scheme, as discussed later, the delay verniers have a minimum time step size equal to one-half the user-programmable resolution.

Table I
System Requirements for the Delay Vernier

Specification	Absolute	Relative	Typical CMOS
Span	16 ns		
Resolution	8-bit or 62.5 ps		
Linearity	<1 LSB		
Skew:			
Temperature	±240 ps/5°C (ambient)	±0.16%/°C	±0.3%/°C
Power Supply	±10 ps/ 50 mV	±0.07%/ 100 mV	±0.8%/ 100 mV
Jitter	10 ps rms	0.03% rms	≈0.1% to 1%
Pulse Symmetry	not required		
Intrinsic Delay	<8 ns		

In Table I and elsewhere in this article, skew is defined as the variation in delay caused by a change of any external influence, including environmental variations. Jitter is defined as delay variation for a series of edges propagating down the line with all external influences constant. The third column in the table shows the skew and jitter specifications as a percentage of the longest time-sensitive timing path on the ACCEL2 chip. This path occurs on the drive side. Since the magnitude of skew and jitter scales with the length of the path, this allows the specifications to be compared with typical performance seen on standard CMOS designs. This data is given in the final column; it is seen that the required skew temperature dependence and jitter performance for the ACCEL2 chip are significantly better than standard CMOS design practices would yield.

Intrinsic delay is simply the delay through the delay vernier with the delay setting at minimum. It is desirable that this delay be as short as possible, since a longer intrinsic delay can complicate the design or compromise the performance of other blocks in the system. However, achieving a shorter intrinsic delay requires greater on-chip power. This is true of other specifications in the table as well; for example, smaller step size requires more power.

Delay Vernier Architecture

The first step in the design of the delay vernier was to select a delay line architecture. A number of delay line implementations have been reported. These include ramped comparators,[2] charge-coupled devices,[3] ECL gate arrays,[1] and multiplexed and tapped delay lines. Resolution, range, jitter, and skew requirements eliminated most of these; the final choice came down to either a binary-weighted multiplexed delay line or a tapped delay line. These two designs are shown in Fig. 4.

Although the multiplexed delay line architecture requires less power and silicon area, it would require the line to be made up of elements of different delays (specifically a binary-weighted series of elements) with inferior skew and linearity performance compared to the tapped line. Because of device mismatches, a binary-weighted multiplexed delay line may have nonlinearities that cannot be corrected by a simple calibration. This is especially possible if there are gaps in the delay-versus-timing relationship because of mismatches. A tapped delay line scheme greatly reduces the effects of device mismatch and essentially avoids these problems entirely. There are other problems with the multiplexed delay line. Low intrinsic delay is difficult to achieve, since for an 8-bit line the minimum delay would need to pass through eight multiplexers. For these reasons, we chose the tapped delay line structure, Fig. 4b.

As shown in the figure, the tapped delay line consists of two types of delay elements: coarse and fine. The coarse elements are calibrated to an identical fixed delay of approximately 2 ns during an initial calibration procedure. An eight-input wired-OR circuit selects the edge at various points along the line, providing a timing adjustment to a resolution of one coarse element delay. The multiplexer is designed so that its propagation delay is nearly identical through all taps. This required putting an extra dummy load on the tap at the end of the line to equalize the capacitive load on all tap inputs. Despite these measures, however, an actual chip will have

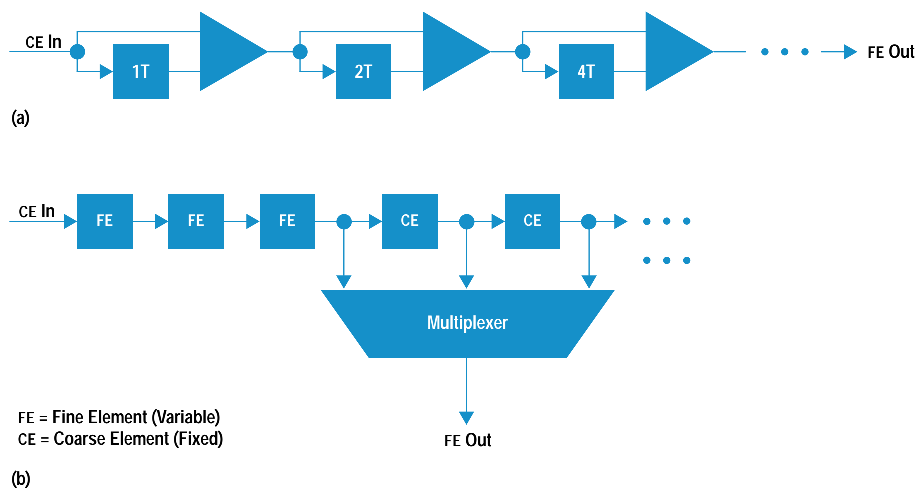


Fig. 4. (a) Multiplexed delay line. (b) Tapped delay line.

Theoretical Approach to CMOS Inverter Jitter

Metal-oxide-semiconductor or MOS technology is known to be inferior in noise performance to bipolar or junction FET (JFET) technology. For example, a CMOS operational amplifier has two to three orders of magnitude worse noise performance than a bipolar or JFET-input operational amplifier. How about jitter? Jitter can be defined as a timing uncertainty or noise. If the device is so noisy, won't the noise affect its jitter performance? We tried to clarify this question by a theoretical calculation.

Major noise sources in a CMOS device are flicker noise and Johnson noise. To the simple CMOS inverter circuit shown in Fig. 1, we applied a noise model and analyzed the thermal jitter.

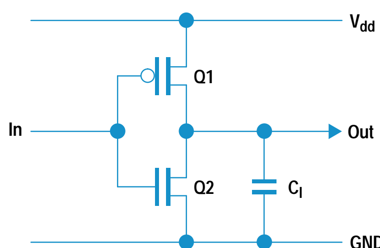


Fig. 1. Simple CMOS inverter with load capacitance C_l.

The jitter is evaluated at the mid-transition point of the output of the inverter (Fig. 2). We assume that the input has already reached the V_{dd} voltage. Q1 completely turns off and Q2 discharges the load capacitance C_l.

Fig. 3 shows the equivalent circuit of the flicker and Johnson noise model, where v_j represents the Johnson noise of the equivalent resistance of Q2 and v_f represents the input flicker noise.

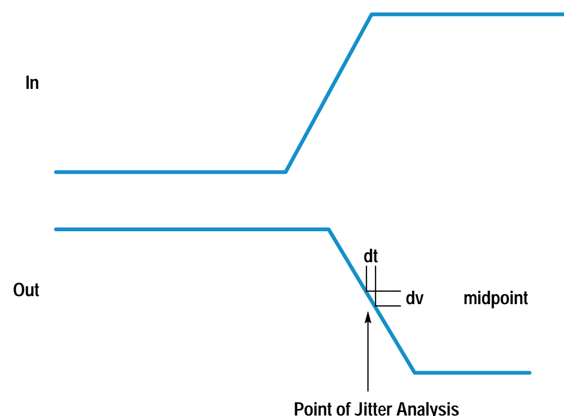


Fig. 2. Jitter evaluation point.

slight mismatches because of manufacturing variations. These can be calibrated out during coarse element calibration, as discussed later.

Finer resolution is provided by three fine delay elements. These have delay that can be varied over approximately a 1-ns range in 31-ps steps by turning on or off internal capacitors, as described later. On the ACCEL2 chip, the same element design is used for the coarse and fine elements. The element has a nominal delay of about 2 ns with the 5-bit digital delay control set at a default value. In the coarse elements, delay variability is used only for calibration.

Two design techniques were key to achieving the required linearity and skew performance: (1) use of essentially identical delay elements throughout the line and (2) use of "thermometer decoding" in control of the line. The benefits of these techniques will be discussed in more detail below. Thermometer decoding is defined as follows: as the input delay setting of the line is increased, delay elements are added to but never removed from the delay path. This guarantees monotonicity of the uncalibrated delay as a function of digital setting and improves uncalibrated linearity, thus making simplified calibration schemes possible. This characteristic is evident in the coarse delay part of the line, but is also employed in the internal structure of the fine elements.

Delay Vernier Element Design

The basic delay element is shown in Fig. 5. It consists of an input inverter, programmable banks of capacitors, and an output inverter. In its quiescent state, the element is presented with a high voltage at the input. An edge propagates through the element when the input voltage undergoes a high-to-low transition. The delay of the edge through the element is determined by the number of capacitors that are turned on and by the bias voltage applied to the gate. When the internal node voltage reaches the switching threshold of the output inverter, the output makes a high-to-low transition, which is applied to the next element in the delay line.

The bias voltage is generated and adjusted with a DAC, and is used to calibrate the delay such that process variations from chip to chip are nulled. The DAC also compensates for delay variations caused by temperature and supply voltage variations.

The capacitors are programmed using a 5-bit digital input. The higher-order capacitor banks are accessed in thermometer-decoded fashion while the lower-order banks are programmed in a straight binary fashion. The breakpoint in this decoding was chosen based on the expected variation in the capacitor elements. This approach minimizes the differential nonlinearity of the delay as a function of the digital capacitor setting. The capacitor array is sized such that the nominal delay through each element is 2 ns. The minimum time step is 31 ps.

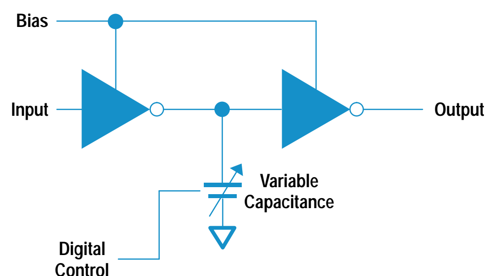


Fig. 5. Basic delay element.

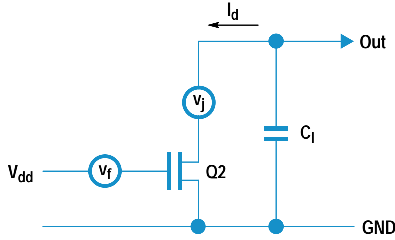


Fig. 3. Noise jitter model.

We calculated the voltage noise V_n and the slew rate dV/dt at the output and estimated the jitter T_{jit} as

$$T_{jit} = \frac{V_n}{dV/dt}.$$

Fig. 4 shows the result of calculations for various gate lengths L and gate widths W. T_{jit} does not exceed 0.3 ps rms even with the worst-case device. A series of 50 such inverters will only produce 2.1 ps rms jitter. While not perfectly accurate, this analysis provides valuable insight.

Another jitter source is coupling noise. This is considered to be a dominant source of jitter in standard CMOS design. In the ACCEL2 design, we avoided subdividing the test period or PCLK signals because subharmonics cause periodic jitter. When the ACCEL2 chip is generating a periodic waveform, all coupling sources are phase coherent with the timing edges. This means we will see the identical coupling waveform at every test cycle and it won't cause any jitter.

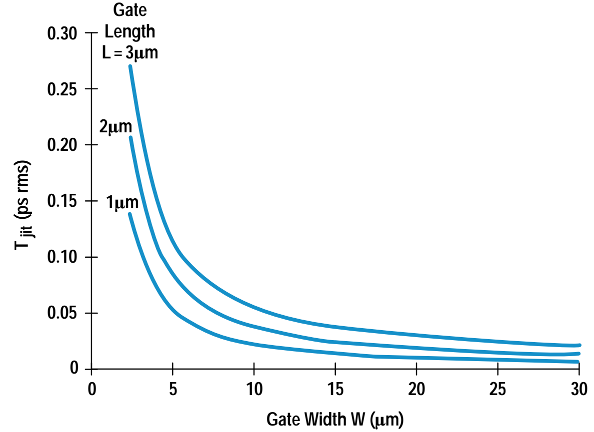


Fig. 4. Calculated CMOS inverter jitter.

Masaharu Goto
Design Engineer
Hachioji Semiconductor Test Division

Support Circuitry

As already discussed, a tapped delay line architecture was chosen for linearity considerations. As seen in Fig. 6a, the delay line contains twelve delay elements and supporting circuitry. Three delay elements are used for fine adjustment of the overall delay. During calibration, the digital settings required to generate a given delay are stored in a look-up table RAM. There is one calibration look-up table RAM per delay vernier. Three elements are used in the fine section to guarantee a span of at least 2 ns corresponding to the delay of one coarse element. The output of the third fine delay element is presented to seven coarse delay elements whose

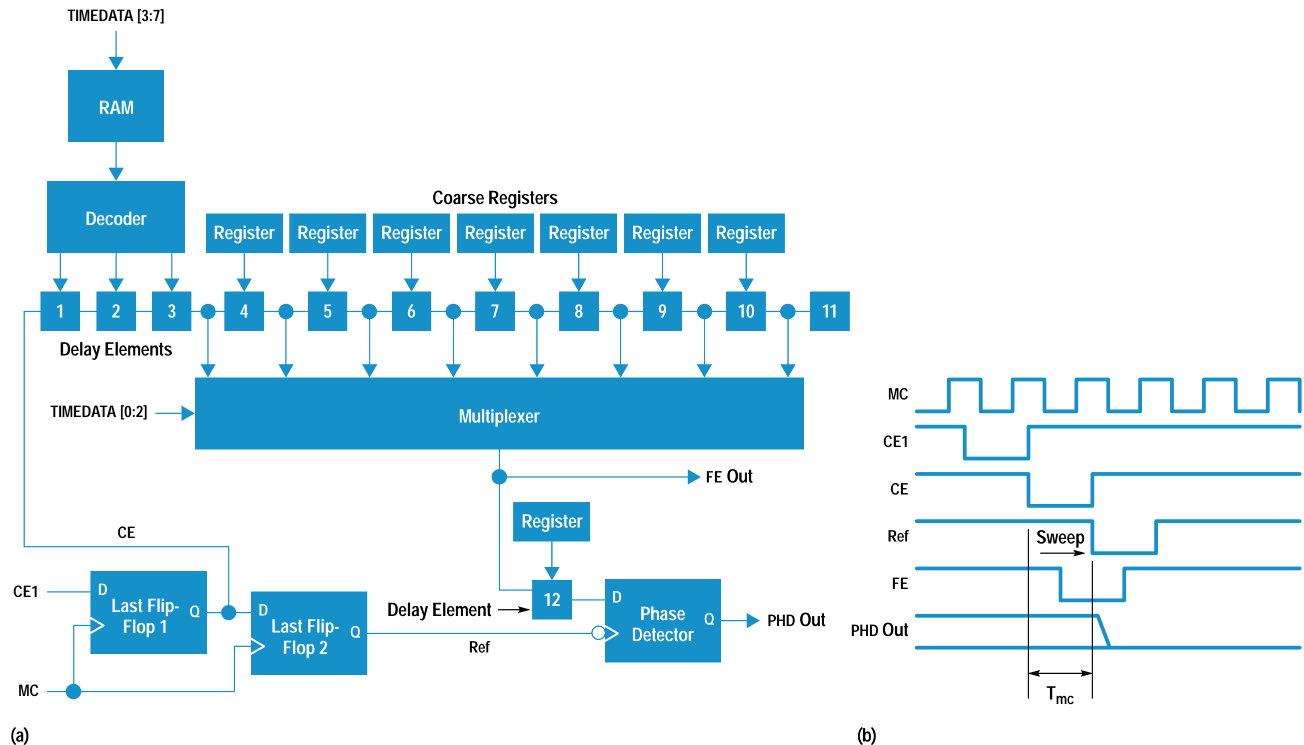


Fig. 6. (a) ACCEL2 delay vernier. (b) Calibration timing waveforms.

digital settings are adjusted during calibration to get a delay as close to 2 ns as possible. A dummy element at the end of the line is used to load the final element.

The delay line is driven by a flip-flop with stabilized propagation delay. This last flip-flop is driven by a so-called coarse edge signal generated by a counter in the digital section. The clock for the last flip-flop is a master clock that is buffered and distributed exclusively to the delay vernier section of the chip. This approach removes jitter on the coarse edge caused by noise generated in the digital section of the chip. To minimize its own contribution to jitter and skew, the last flip-flop is designed to have a short clock-to-output delay.

The multiplexer selects one of eight taps along the delay line to be directed to the output. The multiplexer elements are simple two-input NOR gates whose outputs are connected in a wired-OR arrangement and buffered by a final inverter. This circuit is designed to have minimum propagation delay without unduly loading the basic delay elements. Using this arrangement, the propagation delay varies from some minimum intrinsic delay, T_i, to T_i + 16 ns adjustable in 31-ps increments. Five bits of timing data are used to select one of the 32 fine delay look-up table entries and three bits of timing data are used to control the multiplexer setting. T_i is given by

$$T_i = 3T_{idl} + T_{imux}$$

where T_{idl} is the intrinsic delay of the basic element, and T_{imux} is the delay through the multiplexer. The multiplexer delay in the final design is 1.75 ns and the delay through the basic delay element with a digital setting of 0 is 1.75 ns. Thus the overall intrinsic delay of the line is 7 ns.

Used only during calibration are an additional delay element and a special flip-flop called the phase detector, which is designed for minimum setup time and is used to detect a match between the master clock frequency and the delay through the delay line. The operation of this additional calibration circuitry is described below.

Calibration Scheme

The HP 9493 system uses a high-quality frequency synthesizer as a master clock source. A master clock frequency from 4 kHz to 128 MHz can be programmed in 1-microhertz steps with extremely low jitter and high frequency stability. This translates to subpicosecond master clock period (T_{mc}) programmability. Therefore, we decided to use the master clock period T_{mc} as a timing reference for linearity calibration. A second last flip-flop delays the CE signal by one master clock period to create Ref. Because last flip-flop 1 and last flip-flop 2 are identical and are toggled by the same MC, the time interval between CE1 and Ref is equal to T_{mc}, which we can arbitrarily control from 7.8 ns to 250 μs (Fig. 6b). As described above, the intrinsic delay of the delay vernier is 7.00 ns. Because 7.8 ns is the shortest controllable time interval, we needed to put another delay element after FE. This element adds another 2 ns to the calibration path, resulting in a 9.00-ns minimum delay setting. This is within the range of control of the master clock period T_{mc}. The twelfth delay element is only used for linearity calibration. To prevent an increase in intrinsic delay and timing skew, FE does not go through this element.

The linearity calibration process for each individual delay vernier consists of three parts: bias DAC calibration, fine delay look-up table calibration, and coarse register calibration.

First, all delay elements are programmed to a default value. The bias DAC setting is then calibrated to adjust each element delay to approximately 2 ns. Process variation is roughly calibrated out by this step. Secondly, the fine delay look-up table RAM is calibrated for addresses 0 to 31. At RAM address 0, the master clock frequency is set to 111 MHz (T_{mc} = 9.0 ns) and the RAM data, which is applied to the fine delay elements, is incremented until the delay through the three fine elements equals this value. At the next RAM address, T_{mc} is incremented by 62.5 ps and the RAM data value is again incremented to cause the vernier delay to match the clock period. This is repeated 32 times to calibrate all RAM addresses with 62.5-ps resolution. Thirdly, each coarse delay element is calibrated, starting with the first coarse element. RAM address 0 is selected, the second multiplexer tap is selected, and T_{mc} is set to 11.0 ns, which is 2 ns more than the value used to calibrate RAM address 0. The coarse element register value is incremented until the delay matches this clock period. All the remaining coarse elements are calibrated in a similar manner with the master clock period increased by 2 ns for each successive element. The coarse element calibration compensates for slight variations in the multiplex delay through various taps in addition to calibrating the delay element itself.

The ACCEL2 chip contains a calibration sequencer block called the calibration logger, which supervises the per-pin parallel timing calibration. The calibration logger increments digital timing data for the three phases of calibration described above until the phase detector output of the particular time vernier changes state. At the setting that causes the phase detector to change, the calibration logger stops incrementing the timing data and logs the value of the digital input.

During each calibration operation, the calibration logger can average up to 256 pulses to prevent occasional noise from terminating the measurement prematurely. In addition to calibrating the delay verniers, the calibration logger performs other system calibration and deskewing operations. After all calibration logger operations are completed, the tester controller reads the logged value to get the measurement results. Since all timing vector generator channels in the system can operate in parallel, computer overhead is small and we can perform full linearity calibration of a 256-pin system within 30 seconds.

Dynamic Power Compensation

While the power dissipation of an ECL device remains approximately constant for all operating conditions, a CMOS device changes its power consumption drastically between the static and dynamic states. This is because the power required to charge and discharge internal capacitances when the nodes are toggling at a high frequency is much greater than the standby leakage power. This dynamic power variation is a problem when integrating precision analog circuits and a large amount of digital logic onto a single CMOS VLSI chip. Operation of the analog circuit is often sensitive to temperature, and junction temperature changes caused by

dynamic power variation can be a major source of inaccuracy. The ACCEL2 chip contains 20,000 gates of CMOS logic along with precision delay verniers. The dynamic power variation inherent in the logic cannot be neglected. For a given package thermal resistance, the junction temperature was estimated to vary by 4.5°C because of dynamic power variation. Even with the reduction of delay temperature sensitivity afforded by the custom time vernier design, this variation will unacceptably degrade the system timing accuracy. The dynamic power compensation circuit was developed to solve this problem.

Generally in CMOS logic design, power dissipation is almost proportional to clock frequency. Two on-chip clock networks dominate the ACCEL2 logic operation: the master clock MCLK and the period clock PCLK. From a previous design, we found that the dynamic power of the chip could be reasonably predicted from the frequency of these two clocks. The MCLK frequency is programmed by the tester controller and stays constant during critical operation, so the MCLK dependent power can be calculated by the tester controller. PCLK is an external data input latched by MCLK; it initiates a test period. Because research on customer needs indicated that the ability to change the length of the test period on the fly would be a useful feature, the PCLK frequency can change at any time during critical operations. Therefore, the PCLK dependent power cannot be estimated by the tester controller.

To compensate for dynamic power variation without any increase in total power dissipation, we designed a state machine that roughly monitors the amount of activity in the logic circuitry and dynamically turns an on-chip power compensation heater on and off. Fig. 7a is a conceptual schematic diagram of the dynamic power compensation circuit and Fig. 7b illustrates the principle of operation. When the test system is not in use and MCLK is not running, MCSTAT is reset (MCSTATN = 1). At this time, the X register value is fed to the heater. The X register is programmed by the tester controller as follows:

$$X = P_{pcmax} + P_{mcmax},$$

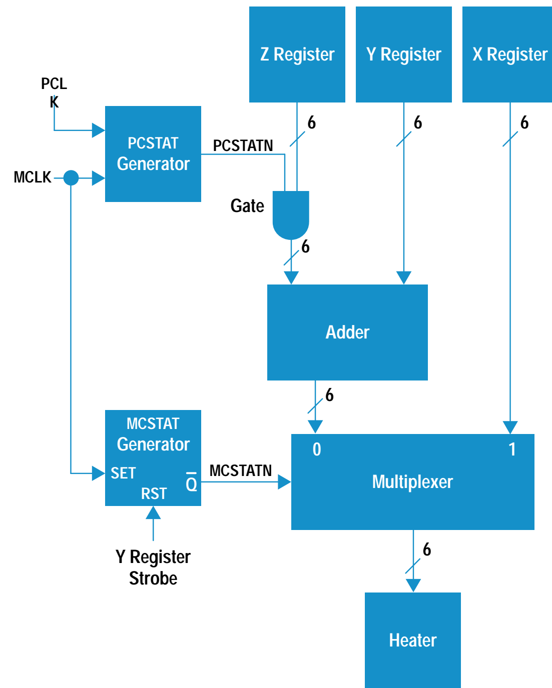
where P_{pcmax} is the PCLK dependent power at the maximum PCLK frequency f_{pcmax} (not shown in Fig. 7b) and P_{mcmax} is the MCLK dependent power at the maximum MCLK frequency f_{mcmax}. At this moment, the dynamic power dissipation of the ACCEL2 chip is zero. Therefore, the sum of the dynamic power and the heater power is simply $X = P_{pcmax} + P_{mcmax}$. The tester controller sets the Y and Z values such that:

$$Y = \left(P_{pcmax} + P_{mcmax}\right)\frac{f_{mcmax} - f_{mc}}{f_{mcmax}}$$

$$Z = P_{pcmax}\frac{f_{mc}}{f_{mcmax}}$$

When MCLK starts running at frequency f_{mc}, MCSTAT detects this condition and MCSTATN becomes 0. Because PCLK is not toggling, PCSTAT is reset (PCSTATN = 1) and Y+Z is fed to the heater. Dynamic power in this state is shown as P_{mcdyn} in Fig. 7b. The sum of the dynamic power and the heater power is equal to $P_{pcmax} + P_{mcmax}$.

When PCLK transitions, PCSTATN becomes 0 for N MCLK cycles, which is equal to the minimum PCLK interval, and then returns to 1. During this period of N MCLK cycles, the Z value is gated off. This decreases the heater power by an amount equal to the dynamic power consumed by a single PCLK cycle. Therefore, the sum of the dynamic power and the heater power remains $P_{pcmax} + P_{mcmax}$ regardless of the PCLK frequency.



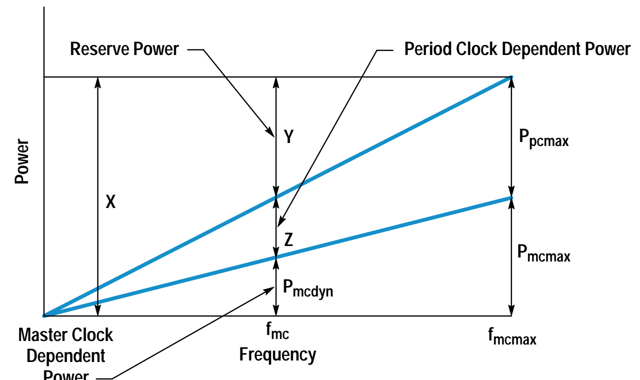


Fig. 7. (a) Dynamic power compensation circuit. (b) Power relationships.

When MCLK is stopped by the tester controller, MCSTAT is reset immediately afterwards so no significant power glitch will occur.

In this way, the total power consumption of the chip is kept constant. This scheme greatly improves system timing accuracy.

Measured Delay Vernier Performance

Fig. 8 shows the raw (uncalibrated) integral nonlinearity of the three fine elements measured on the ACCEL2 chip. The

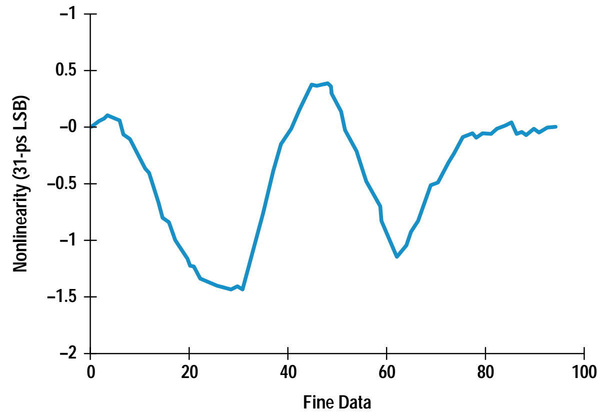


Fig. 8. Uncalibrated fine delay element linearity.

fine element nonlinearity is approximately +1.5 LSB of 31-ps raw resolution.

Fig. 9 shows calibrated delay vernier linearity. This curve covers the entire digital input range of 0 to 255, with a corresponding delay range of 0 to 16 ns. The nonlinearity is expressed in terms of the system LSB of 62 ps. As the curve shows, the linearity calibration guarantees less than ±1 LSB of integral nonlinearity in the system timing resolution over the entire delay range.

Jitter measurement was done using an HP 54121T digitizing oscilloscope with the delay line set at the maximum value of 16 ns. The measured jitter was 3.3 ps rms. Removing the intrinsic jitter of the HP 54121T (evaluated to be 1.2 ps rms) resulted in an estimated ACCEL2 jitter of less than 3.1 ps rms. This is about 0.01% of the path delay, which is one to two orders of magnitude better than typical digital designs and better than the specification by a factor of three (see Table I).

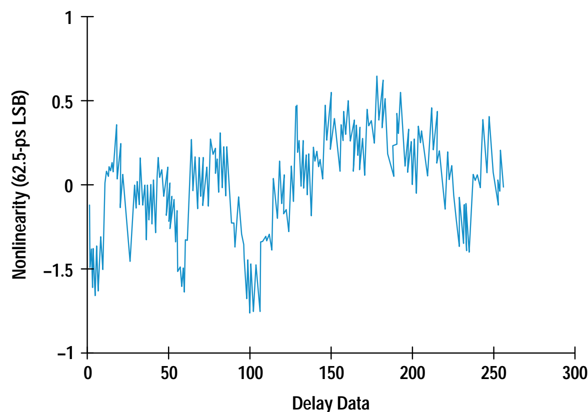


Fig. 9. Calibrated delay vernier linearity.

Table II shows the measured temperature coefficient of the propagation delay through the delay vernier circuitry and CMOS formatter block. As the table shows, the ACCEL2 delay vernier is about three times better than the custom CMOS formatter circuit. This temperature stability performance is equivalent to bipolar time verniers. In the ACCEL2 chip the overall temperature coefficient was measured as 30 ps/°C.

Table II
Temperature Coefficient of Propagation Delay

CMOS Formatter	0.15%/°C of path delay
Delay Vernier	0.058%/°C of path delay
Total Critical Path	30 ps/°C

Measurements were also made of the power supply dependence of the delay of a critical timing path . The propagation delay was extremely stable as the V_{dd} voltage changed.

Conclusion

Integration of delay verniers with formatter logic in the custom VLSI chip ACCEL2 was the key to achieving a low-cost, low-power LSI test system design. By moving the delay lines on-chip, the cost and power of the timing system were reduced by nearly an order of magnitude while at the same time providing ECL-equivalent timing performance.

Acknowledgments

The design was accomplished by a cooperative effort of the Integrated Circuits Business Division Fort Collins Design Center and the Hachioji Semiconductor Test Division. We would like to thank Albert Gutierriz and Chris Koerner for designing a major part of the delay vernier. Gary Pumphrey of the Colorado Springs Division contributed key ideas used in the design of the delay element. Keith Windmiller managed the project and provided guidance and direction. Barbara Duffner designed the high-performance CMOS formatter and kept track of many other details during the course of the project. Koh Murata and Kohei Hamada designed the surrounding system of the chip. We would also like to thank Larry Metz and Charles Moore for their invaluable consulting for the advanced CMOS analog design.

References

1. T.I. Otsuji and N. Naumi, "A 3-ns Range, 8-ps Resolution Timing Generator LSI Utilizing Si Bipolar Gate Array," *IEEE Journal of Solid-State Circuits,* Vol. 26, no. 5, May 1991, pp. 806-811.

2. T. Ormond, "Delay Lines Take on Timing Tasks," *EDN Magazine,* December 1991, pp. 108-112.

3. C.F.N. Cowan, J.W. Arthur, J. Mavor, and P.B. Denyer, "CCD Based Adaptive Filters: Realization and Analysis," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* Vol. ASSP-29, no. 2, April 1981, pp. 220-229.

Real-Time Digital Signal Processing in a Mixed-Signal LSI Test System

In test subsystems based on digital signal processing, the HP 9493 test system emulates the analog and digital signals of the device under test, thereby reducing test time and increasing test coverage compared to a memory-based test system.

by Keita Gunji

Complex mixed-signal integrated circuits are used more and more frequently to realize many complex functions needed for applications such as telecommunications, servo control, and image processing. Testing of these ICs is much more difficult than for traditional ICs because of the integration of complex functions and synchronized interactions between blocks. Conventional IC testing methods cannot provide both high test throughput and high test quality for these devices. At-speed signal processing by synchronized distributed subsystems is required to perform functional testing of complex mixed-signal devices. Also required is a programming style that is straightforward and allows flexible control of signal and data flow in the test subsystems.

DSP-based LSI testers can be classified as having one of two types of architecture according to the location of the signal processing units (SPU): centralized or distributed.[1] In the distributed type, SPUs are distributed among the individual subsystems. Usually, each SPU cannot manage a large amount of data and the processing speed of each SPU is slower than that of a centralized SPU. Programming is more difficult because the SPUs are not strongly coupled to the CPU, so the control layer between the CPU and the SPUs is more complicated. On the other hand, the distributed type makes it possible to process data in the subsystems and enables at-speed emulation of IC functions, which is required to test mixed-signal devices.

To meet the needs of complex mixed-signal testing, the HP 9493 mixed-signal LSI test system is based on distributed digital signal processors (DSPs). In each test subsystem is a Motorola DSP96002 chip, which runs at 32 MHz and performs 32-bit floating-point operations. The DSP modules are connected by data transmission paths for data sharing. A flexible programming language directs the DSPs to generate and analyze test signals.

The HP 9493 architecture allows digital and analog signal processing completion times on the order of microseconds to ensure the quality and throughput of dynamic signals and data operations for the device under test (DUT). Precise timing synchronization of all subsystems makes it possible to distribute operations among the DSP modules.

Compared to a memory-based test system, the HP 9493's distributed DSP architecture reduces test time and increases test coverage for complex mixed-signal devices. To generate an analog waveform, the conventional memory-based system must compute and store discrete time data for the entire waveform before generating it. This takes time and may require so much memory that the functions of a complex device may not be tested completely because of the memory cost. In contrast, the DSP-based HP 9493 prepares the discrete time data in real time using a much smaller amount of waveform memory.

Architecture

Fig. 1 shows the HP 9493 system architecture, which allows synchronous and asynchronous control of the analog and digital subsystems and the distributed DSP in each subsystem. This distributed architecture consists of a tester controller, which controls all test subsystems, a master clock subsystem, two test heads, and test subsystems. The test subsystems include digital subsystems, which generate and fetch digital patterns and signals, arbitrary waveform generator subsystems, and waveform digitizer subsystems.

In each test subsystem are a DSP module, a sequencer, and a memory block. Each DSP module has a data transmission path. The DSP modules operate on data or signals according to instructions loaded from the tester controller, and the start and stop operations are synchronized by the master sequencer or tester controller. The master sequencer precisely synchronizes the timing of the other sequencers.

The DSP96002 manages the memory block. Data can also be stored in high-speed memory in the DSP module to reduce memory access time, which increases the processing speed of the DSP96002. The sequencer in each subsystem controls address generation and the timing of data fetch or generation between the memory block and the signal input or output for memory-based testing.

The advantages of this architecture are real-time signal emulation and shared signal processing. Test throughput is more than double the capability of conventional LSI testers, and ICs can be evaluated by at-speed functional tests. Data can be transmitted between DSP modules without CPU intervention, which allows synchronous and concurrent operation and sharing of signals from the DUT.

The DSP module in the digital subsystem can fetch and generate up to 32-bit digital test data. The test data is directly connected to the data bus of the DSP96002. Data can be

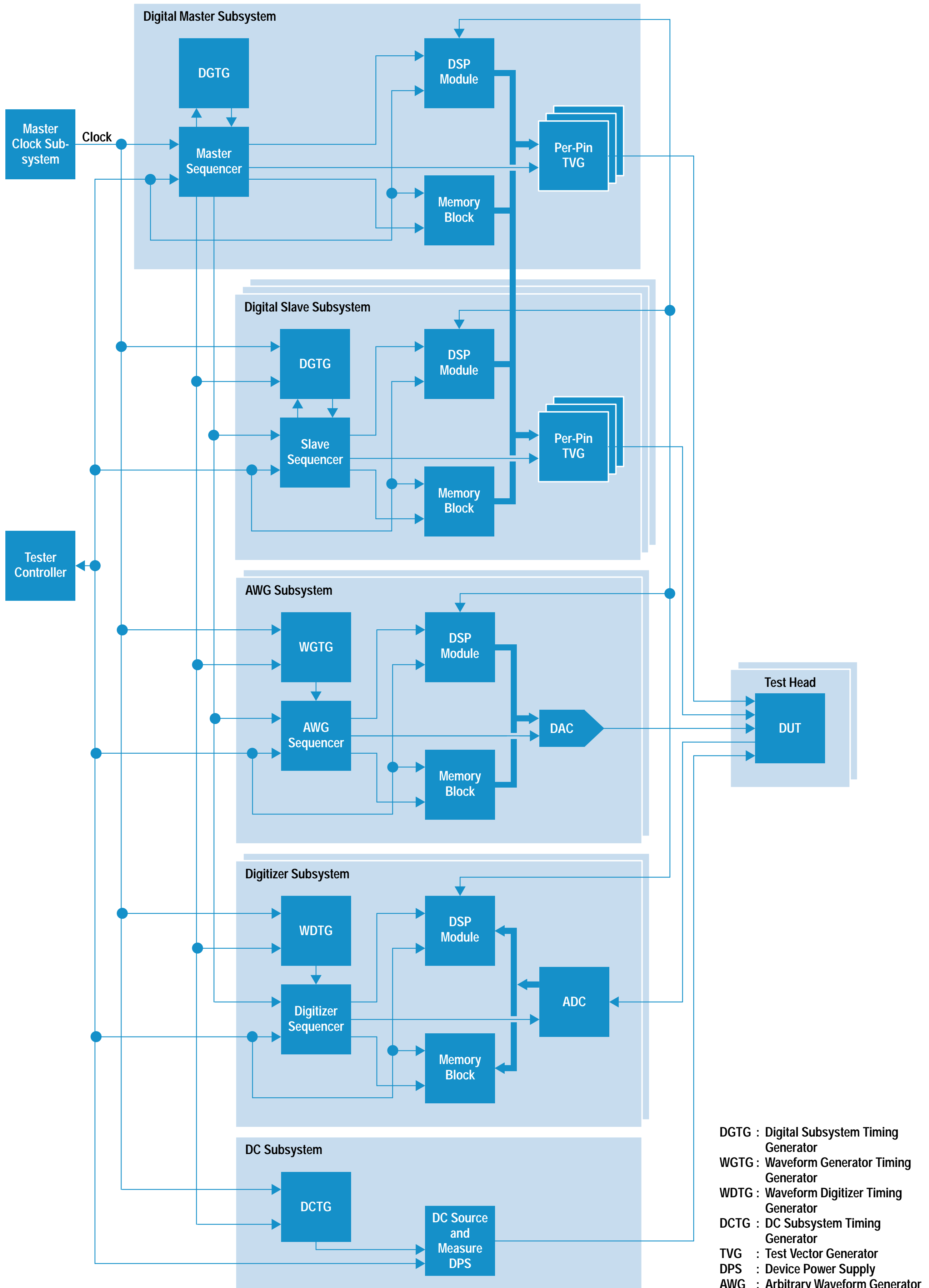


Fig. 1. The HP 9493 mixed-signal LSI test system architecture puts a local DSP module in each test subsystem.

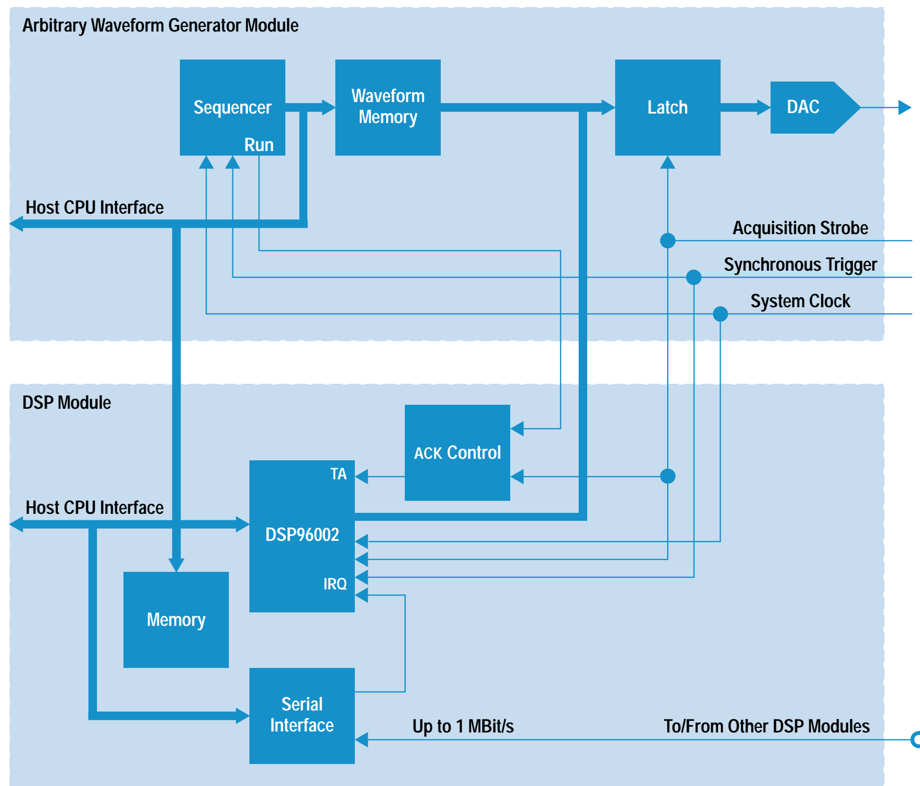


Fig. 2. Diagram of the DSP module and arbitrary waveform generator interface.

transmitted between the master and slave subsystems on a per-pin basis. This is necessary for high-pin-count devices or multidevice testing. The number of pins can be expanded to 256. The analog subsystems are connected to another data transmission path to enable real-time data transmission across subsystems while testing.

The DSP module in the waveform digitizer fetches and processes real-time or buffered sampling data, and can transmit data to other subsystems and the digitizer memory block. Real-time processing and a memory interface allow the capture of DUT responses or signals without restarting the test sequence. This is useful to emulate functions such as ISDN activation. Conventionally, another ISDN device is used to generate the signals required for testing this function, but this method does not have the flexibility to accommodate changing test specifications. Emulation of the activation sequence also allows complete control and synchronization with the tester during the test sequence, which is difficult with an actual IC. A memory-based test system would require over 20 Mbytes of waveform memory to emulate this function. In contrast, the HP 9493 can do it with a maximum waveform memory size of only 2 Mbytes.

The DSP module in the arbitrary waveform generator fetches data from the waveform memory or another DSP module. Waveform data can be generated using a convenient waveform editor, loaded from the CPU directly, captured by the digitizer subsystem, or fetched from the digital subsystem.

DSP Module

Fig. 2 shows a block diagram of the DSP module. Synchronous lines controlled by the master sequencer are connected to the DSP's interrupt and the subsystem's acquisition strobe, allowing synchronous or coherent signal processing across subsystems. The synchronous trigger lines start operations of the subsystems and DSPs.

A test program in the DSP module must run an initialization process for any variables and global setup and cannot start with the timing of this trigger line. For complete synchronization, the DSP96002 is controlled to wait for an acquisition strobe at its acknowledge input before doing any data sampling or generation. DSP data acquisition is synchronized with the DUT by this acknowledge line. The acknowledge line is connected to the sequencer and the acquisition strobe to enable full synchronization of all DSP modules within one DSP cycle.

For digitizing, initialization starts before real-time data sampling. The acknowledge control waits for the acquisition strobe of the digitizer before moving data from the digitizer to the DSP. After the initialization of a test program, if the user starts a digitization, the DSPs get data from the beginning of sampling synchronized with the strobe. Synchronization is required to analyze vector data such as the I (in-phase) and Q (quadrature) outputs of a $\pi/4$ DQPSK device accurately.

For waveform generation, the sequencer halts data movement from the DSP while the sequencer is generating data to the DAC. It can switch data between the DSP and the waveform memory during an acquisition cycle. The waveform in Fig. 3 shows an example.

The acquisition strobe timing of all of the digital and analog subsystems is accurately managed at the DUT pins. Synchronization accuracy on the order of 100 ps is maintained.

Fig. 3. Synchronous waveform switching. The highlighted window is an editor for programming waveform generation. The instruction BNDSP Start,DSPON switches data to the DAC. The waveform is displayed in the waveform editor window and shows the switching between the ramp from the waveform memory and the multitone wave from the DSP. The upper window is a pattern editor for programming a test sequence, the start and stop of the synchronous trigger, and digital patterns and formats. START in the SCE 1 column starts the arbitrary waveform generator sequencer and the DSP.

The data transmission paths allow data transmission between subsystems for synchronized interaction, and are also used to synchronize the distributed DSPs with an accuracy of one DSP clock cycle. These paths conform to the RS-422 standard. The test subsystem grounds are isolated to reduce noise transmission.

Software

The usual problem with a distributed system is the complexity of the control layer from the CPU and the fact that the signal processing function cannot be controlled directly by the CPU. To test various chips, the test system software should be flexible enough to execute any DSP program. Easy programming is required for the inputs and outputs of the test subsystems. The HP 9493 software hides the complexity of hardware control, and uses a programming style based on signal analysis using data flow diagrams, which consider signals and data flows in the test system. This programming style separates system hardware control and low-level DSP programming so that DSP algorithms can be developed independently of the tester.

Modular Operation Library. Conceptually, the test system hardware can be represented by a data flow diagram as shown in Fig. 4. Each operation shown in Fig. 4 can be described as shown in Fig. 5. Each operation has inputs and outputs, and the user can switch the I/O and execute the same operation using different subsystems. The HP 9493 test plan language controls the loading of the operation and the I/O connections. Using this conceptual model, the complexity of hardware control can be hidden by the software, so that programmers only need to define the input and output relationships between operations and test subsystems.

For operations that are signal processes such as filtering, algorithms can be programmed using the DSP's assembler and compiler without considering the hardware subsystems.

Test Plan Language. A test plan language is supplied to create test sequences for the HP 9493 test system (see Table I). Programming is based on the data flow diagram.

Table I
HP 9493 Test Plan Language

Function	Parameters	Description
LOAD-MOL	PIN LABEL MOL	Selection and loading of an operation library
SET-MOL-MODE	PIN OPERATION-MODES	Definition of the operation mode of the DSP module
SET-MOL-IO	LABEL INPUT OUTPUT	Input and output definition of the operation library
SET-MOL-ARG	LABEL ARGUMENTS	Setting of the arguments of the operation library
READ-MOL	PIN NUM-OF-ARRAY RESULT-ARRAY	Reading of output from the operation library

The test plan language can be used to execute any function and makes it easy to create functional tests using the DSP capabilities in the HP 9493 test system.

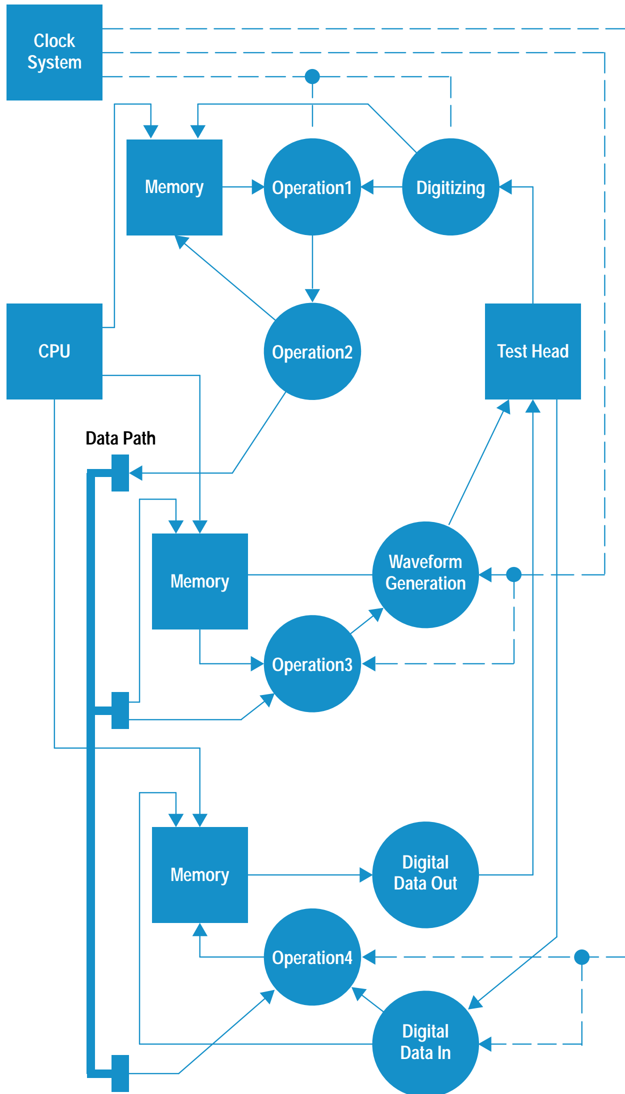


Fig. 4. Data and signal flows in the test system. This diagram shows the top level of the tester architecture. The solid lines indicate data flows from the CPU. The dashed lines are synchronous control flows from the clock system.

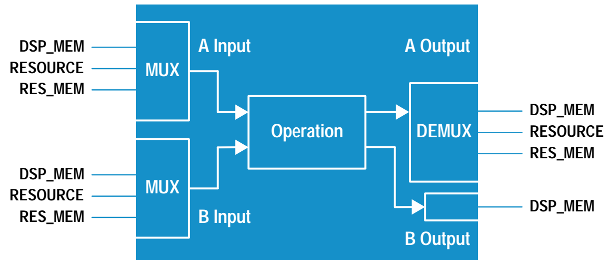


Fig. 5. The modular operation library (MOL) conceptual module of one of the operations in Fig. 4. An operation is an object used to realize signal processing. It is an object of the DSP assembler, and is loaded in the DSP and executed to process any input data to a MOL. The processed result will be stored in DSP memory or directly output. There are five types of input and output for a MOL. DSP_MEM: A DSP has memory for storing calculated or downloaded data. The memory space is allocated as input or output. RESOURCE: A resource is a test subsystem—the digitizer, the arbitrary waveform generator, or a digital test vector generator. This I/O line is used to input measurement data to a MOL, or to generate and output calculated data from a MOL directly to the subsystem. RES_MEM: A module memory is arbitrary waveform generator memory, digitizer memory, or digital capture memory. This I/O line is used to input data from the module memory or output data to it.

Conclusions

The HP 9493 mixed-signal LSI test system contains high-performance digital signal processing modules in each test subsystem. This allows data processing with synchronous interaction for at-speed functional testing of mixed-signal devices. A programming language for test library development is provided by the system software. This helps develop and execute functional tests of various mixed-signal devices.

Acknowledgments

The authors would like to thank project manager Masaki Yamamoto, Noriaki Hiraga, Masahiro Kimoto, and Yoshiyuki Bessho for their contributions to the software construction, Tomoaki Kobayashi and Hiroyuki Shimizu for the hardware design of the DSP modules, Toshichika Ohmori for his diagnostics design, Heihachi Sato, Shinichi Saito, and Yasuharu Harada for their mechanical support, and Masahiro Yokokawa, Masato Tada, and Kiyoyasu Hiwada for their management support. Thanks to all the members of the HP 9493 project who helped integrate the test subsystems.

Reference

1. M. Mahoney, *DSP-Based Testing of Analog and Mixed-Signal Circuits,* IEEE Computer Society Press.

Vector Error Testing by Automatic Test Equipment

Mixed-signal testers are frequently used as specialized automatic test equipment in various test applications. The real-time digital signal processors in the HP 9493 mixed-signal LSI test system can perform complex tests for next-generation telecommunication devices.

by Koji Karube

In the next generation of wireless communications systems, high-quality telecommunication and compact equipment will be achieved by using new highly integrated, multifunctional, mixed digital and analog devices. Mixed-signal testers like the HP 9493 can test these devices. Many of the newer telecommunication systems use complicated modulation methods such as π/4 DQPSK (differential quadrature phase shift keying), which creates specialized waveforms that are very difficult to analyze. With its built-in digital signal processors (DSPs), the HP 9493 can also be used to solve some of these difficult signal analysis problems.

In general, a mixed-signal LSI tester is not an instrument for measurement but serves as automatic test equipment in the production area. The most important requirement is how quickly failing devices can be rejected from a large number of passing devices. Good repeatability, which is a function of system stability, and usability are also required for high productivity.

Many of the tests performed by mixed-signal testers do not have precise specifications because of the complexity of the device under test. Inside a mixed-signal device are many digital and analog signals, and digital signals may corrupt analog signals and vice versa. As a result, some tests have to be guaranteed statistically or experimentally. The measurement of the vector error, one of the most important test parameters for π/4 DQPSK devices, belongs to this category, so it is not an unfamiliar type of test for a mixed-signal LSI test system like the HP 9493.

We have developed a test application to measure the vector error. The test achieves high throughput and good repeatability.

Roll-off Filter Design

Fig. 1 shows the simplified block diagram of a transmitter baseband device. This device has four signal processing blocks. First, the serial-to-parallel converter converts serial data to two-bit width. The next block encodes the data differentially and maps it into the coordinates of the signal space according to the signal space diagram shown in Fig. 2. The final roll-off filters are a kind of low-pass filter called "root raised-cosine" filters, defined by:

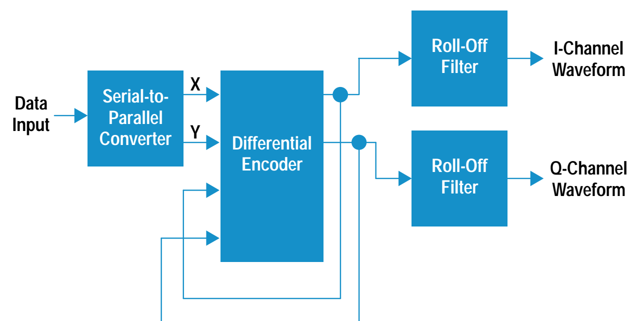
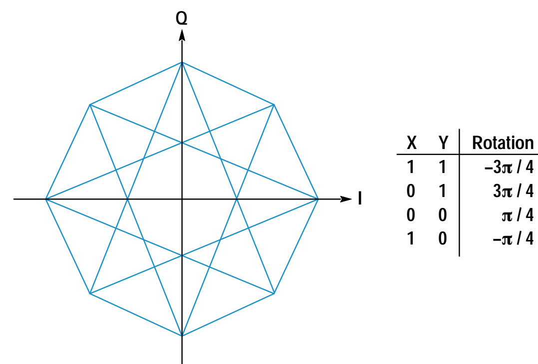


Fig. 1. Baseband transmitter device for π/4 DQPSK modulation.

$$H_\alpha(f) = \begin{cases} 1 & \text{for } 0 < f < \dfrac{1-\alpha}{2T} \\ \sqrt{\dfrac{1 - \sin[(fT - 0.5)\pi/\alpha]}{2}} & \text{for } \dfrac{1-\alpha}{2T} \le f \le \dfrac{1+\alpha}{2T} \\ 0 & \text{for } \dfrac{1+\alpha}{2T} < f \end{cases}$$

where f is the frequency in hertz, T is the data symbol period in seconds, and α is the roll-off coefficient.

The π/4 DQPSK receiver also has root raised-cosine filters such that the overall filter characteristic is raised-cosine with an impulse response that results in a state of no intersymbol



X	Y	Rotation
1	1	−3π / 4
0	1	3π / 4
0	0	π / 4
1	0	−π / 4

Fig. 2. Signal space diagram of π/4 DQPSK modulation.

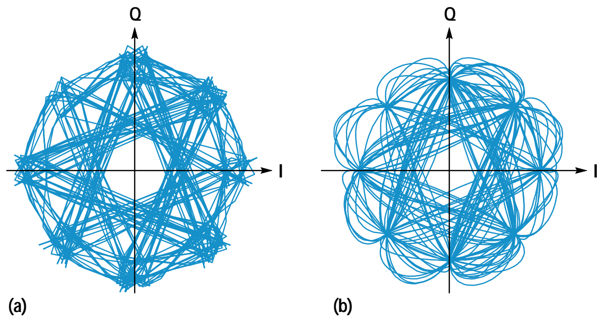


Fig. 3. (a) I-Q signal without filters. (b) I-Q signal with root raised-cosine filters.

interference, as shown Fig. 3b. For this test the receiver filters are emulated in the HP 9493 test system using the digital signal processing capabilities. To obtain the required impulse response, we use 8-symbol-deep FIR (finite impulse response) filters, where 8-symbol-deep means that the duration of the filter impulse response corresponds to eight data symbol times. The design employs virtual oversampling[1] and multirate techniques[1] to achieve high-resolution delay adjustment and high throughput.[2]

Symbol Timing Extraction

The Research and Development Center for Radio Systems in Japan specifies the vector error as follows.[3] The ideal transmitted signal after final filtering is:

$$S(k) = S(k-1)e^{j(\pi/4+B(k)\pi/2)}$$

where B(k) is defined by the following table.

X_k	Y_k	B(k)
0	0	0
0	1	1
1	1	2
1	0	3

The actual transmitted signal after final filtering is:

$$Z(k) = [C_0 + C_1\{S(k) + E(k)\}]W^k$$

where $W = e^{dr+jda}$ for phase offset da (radians/symbol) and amplitude change dr (nepers/symbol)

C_0 = arbitrary complex constant representing the offset of the origin caused by imbalance of the quadrature modulator

C_1 = arbitrary complex constant determined by the phase and power of the transmitter

$E(k)$ = vector error.

Then the sum of the squared vector errors is:

$$\sum_{k=min}^{max} |E(k)|^2 = \sum_{k=min}^{max} \left|\left\{\left[Z(k)W^{-k} - C_0\right]/C_1\right\} - S(k)\right|^2.$$

System designers attempt to minimize this error by selection of C_0, C_1, and W.

To calculate the vector error, C_0, C_1, and W must be determined. For a baseband transmitter device, the frequency offset described by W is zero because the I and Q signals are observed without quadrature modulation that shifts the signal frequency. The offset of the origin described by C_0 can be obtained easily by statistics. To determine C_1, a

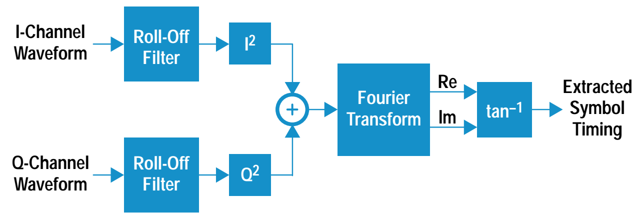


Fig. 4. Conventional method of symbol timing extraction.

method of extracting the phase of the signal (symbol timing) has to be designed. In concrete terms, the function of symbol timing extraction is to find the eight convergence points of the filtered I-Q signal shown in Fig. 3b.

The conventional method of extracting symbol timing is envelope detection of the squared signal as shown in Fig. 4. The phase extracted by this method is a little different from the "best phase" because this method observes only a narrow bandwidth around one-half the symbol frequency and ignores the group delay of the actual devices. Fig. 5 shows the timing error for this method of symbol extraction as a function of the calculated vector error, obtained by simulation. According to this figure, timing error must be within 10 ns to achieve repeatability within less than 0.1%. Therefore, instead of the envelope detection method, we adopted a specialized search method that finds the actual minimum vector error.

Implementation

In a $\pi/4$ DQPSK receiver, the extracted symbol timing is fed back to the digitizer clock inputs. In the HP 9493, this is simulated by using adjustable-delay filters in front of the signal processing.

Fig. 6 shows the simplified block diagram of the vector error test. The test uses two 16-bit digitizers to digitize the I and Q signals independently and uses two sets of real-time DSPs to process both signals at the same time. Each digitized signal with 16-bit resolution is transferred into the roll-off filters and processed by one of the 8-symbol-deep FIR filters. In effect, there are 512 of these filters, each having a slightly different delay. The delay is selected by feedback from subsequent blocks. Next, gain error and offset are adjusted and then

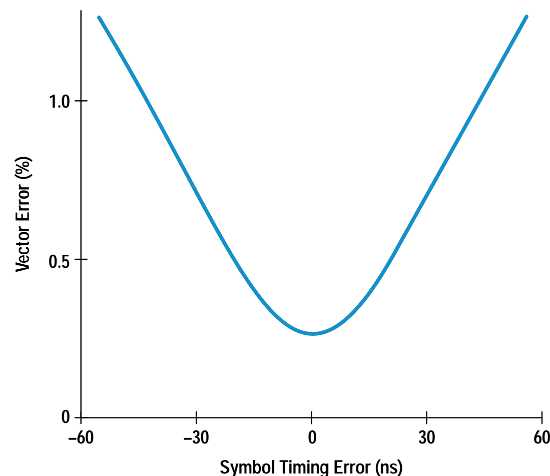


Fig. 5. Symbol timing error as a function of calculated vector error.

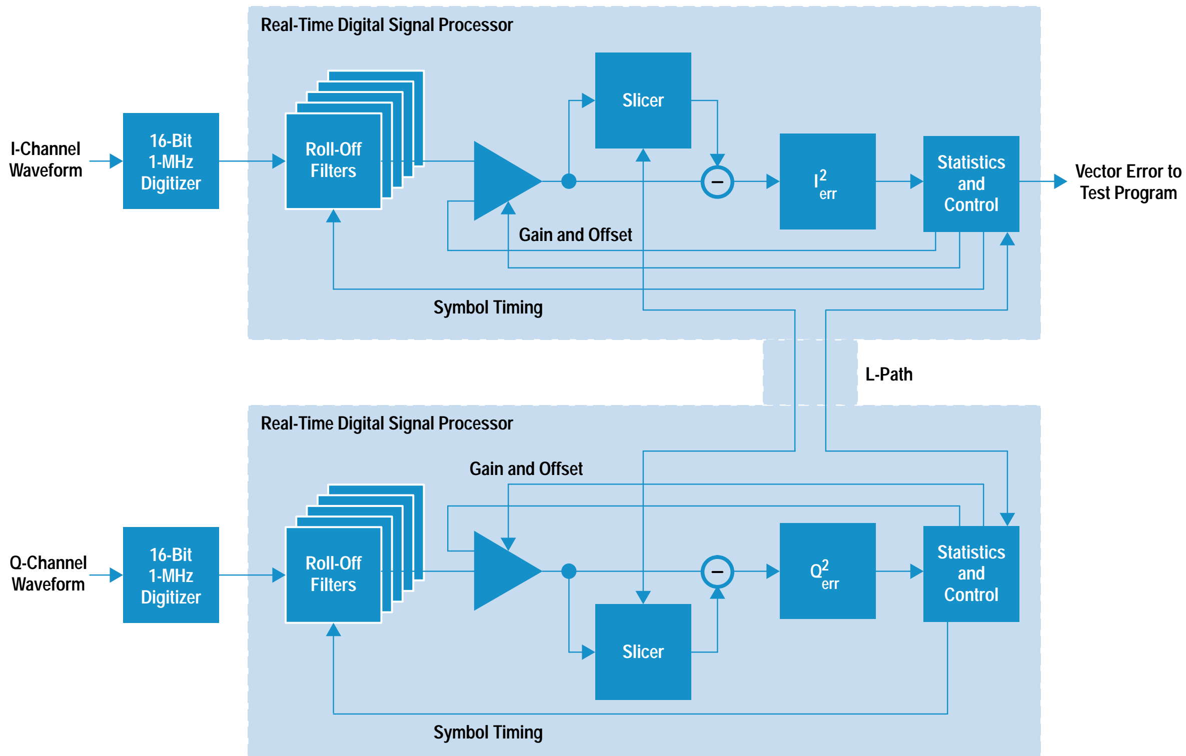


Fig. 6. Block diagram of the HP 9493 vector error test.

squared vector errors are calculated by comparing the actual symbol locations in the I-Q plane with the ideal locations, which are determined by the slicers. The statistics and control block searches for the best symbol timing to minimize the vector error by changing the roll-off filter delay, which changes the symbol timing.

The roll-off filters are designed to have a high virtual conversion frequency of 2048 times the symbol frequency, which corresponds to about 2.5-ns delay resolution in the HP 9493.

Analog Timing Skew

Group delay in the band from zero to the baud rate causes synchronization error. Analog timing skew between digitizers also directly reduces the accuracy of the vector error measurement. In the HP 9493 test system, the effect of timing skew is eliminated by regular calibration, so the user need only make sure that the cable lengths of the I channel and the Q channel are same on the DUT board.

Acknowledgments

The author would like to thank Mamoru Tanba for his benchmark test. The author would also like to thank all members of the application engineering group at Hachioji Semiconductor Test Division for their great cooperation.

References

1. R.E. Crochiere and L.R. Rabiner, *Multirate Digital Signal Processing,* Prentice Hall, 1983.
2. K. Karube, et al, "Advanced Mixed-Signal Testing by DSP Localized Tester," *Proceedings of the IEEE 1991 International Test Conference,* pp. 1055-1060.
3. *Personal Handy Phone System Standard,* RCR STD-28, Research and Development Center for Radio Systems, 1993.

High-Frequency Impedance Analyzer

A new one-port impedance analyzer measures high-frequency devices up to 1.8 GHz. Using a current-voltage method, it makes precise measurements over a wide impedance range. A special calibration method using a low-loss capacitor realizes an accurate high-Q device measurement. Many types of test fixtures are introduced because they are a key element in any test system.

by Takanori Yonekura

In research and development, component qualification, and RF and digital manufacturing, there is increasingly a need to make impedance measurements on chip components, such as chip inductors, capacitors, varactor diodes, and p-i-n diodes, and on other surface mount devices. Often the capacitances and inductances are very small and have impedances much greater or much less than 50 ohms at the operating frequencies. Traditionally, vector network analyzers are used to measure impedance in the RF range, but they are limited to measuring impedances near 50 ohms.

The new HP 4291A RF impedance analyzer (Fig. 1) is designed for passive surface mount device testing and material analysis at frequencies from 1 MHz to 1.8 GHz. Using an RF current-voltage measurement technique, it provides improved measurement accuracy over a wide impedance range, making it possible to test components at operating frequencies and to evaluate non-50-ohm components more accurately.

New surface mount component test fixtures for use with the new analyzer save time and eliminate the need for custom fixtures. These fixtures and the analyzer's built-in calibration and compensation routines ensure measurement accuracy.

This paper is a revised and updated version of reference 1 and is published here with permission of the Electronic Components Institute Internationale Ltd.

The analyzer's firmware provides direct impedance reading, frequency-swept measurements and many advanced functions such as equivalent circuit analysis, limit lines, and markers.

General Impedance Measurements

A general impedance measurement schematic using two vector voltmeters is shown in Fig. 2. In this case, the true impedance (Z_x) of a device under test (DUT) is determined by measuring the voltages between any two different pairs of points in a linear circuit.

$$Z_x = K_1 \frac{K_2 + V_r}{1 + K_3 V_r}, \quad (1)$$

where K_1, K_2, and K_3 are complex constants and V_r is the voltage ratio V_2/V_1.

There are three unknown parameters related to the circuit in equation 1. Once we know these parameters, we can calculate the impedance of the DUT from the measured voltage ratio $V_r = V_2/V_1$. The procedure that estimates these circuit

Fig. 1. The HP 4291A RF impedance analyzer measures impedances of components and materials at frequencies from 1 MHz to 1.8 GHz. New surface mount test fixtures and built-in calibration and compensation routines ensure accuracy.

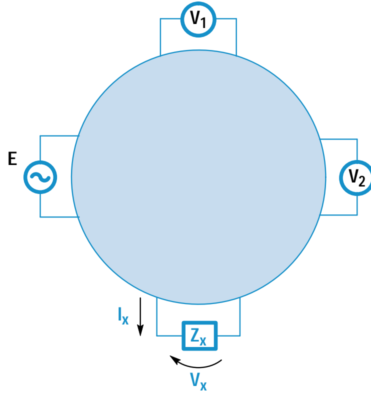


Fig. 2. General schematic for impedance measurement using two vector voltmeters.

parameters is called *calibration* and one method is *open-short-load* (OSL) calibration. Calculation of Z_x from the measured voltage ratio V_r according to equation 1 is called *correction*.

Transducers

We call a linear circuit such as the one in Fig. 2—one that relates a signal source, two vector voltmeters, and a DUT—a *transducer*. Transducers are the key element in impedance measurements. Two types of transducers of interest here are the directional bridge and the transducer in a current-voltage (I-V) method. Let's compare these in terms of their sensitivity to gain variance in the vector voltmeters in Fig. 2.

Directional Bridge. Directional bridges (see Fig. 3) are used in many network analyzers, mainly to measure impedances near 50 ohms. In this case, the bilinear transformation is:

$$Z_x = R_o \frac{1+\Gamma}{1-\Gamma}, \tag{2}$$

where $\Gamma = (Z_x - R_o)/(Z_x + R_o)$ is the reflection coefficient, $V_r = V_2/V_1 = (-1/8)\Gamma$, and $R_o = 50$ ohms is the characteristic impedance. The parameters in equation 1 are $K_1 = -8R_o$, $K_2 = -1/8$, and $K_3 = 8$.

Now assume that the vector voltmeters in Fig. 3 are not ideal but have some gain variance. The measured voltages V_1 and V_2 and the calculated impedance Z_x are:

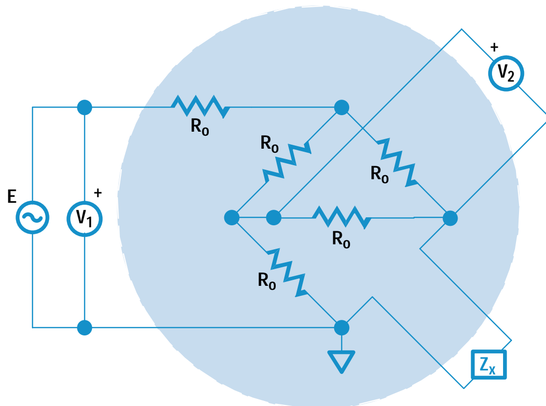


Fig. 3. Directional bridge circuit.

$$V_1 = E\alpha_1$$

$$V_2 = (-1/8)E\Gamma\alpha_2$$

$$Z_x = R_o(1+\Gamma)/(1-\Gamma),$$

where α_1 is the gain of vector voltmeter 1, α_2 is the gain of vector voltmeter 2, $\Gamma = -8V_r\alpha_r$ is the measured reflection coefficient, $V_r = V_2/V_1$ is the voltage ratio, and $\alpha_r = \alpha_1/\alpha_2$ is the ratio of the voltmeter gains.

We define the calculated impedance sensitivity S to the voltmeters' gain variance as follows:

$$S = \frac{\delta Z_x/Z_x}{\delta\alpha_r/\alpha_r}. \tag{3}$$

This sensitivity can be considered as the inverse of the magnification of gain variance. The smaller S is, the smaller the error in the calculated impedance.

For the directional bridge, equation 3 is:

$$S = \frac{\delta Z_x}{\delta\Gamma}\frac{\delta\Gamma}{\delta\alpha_r}\frac{\alpha_r}{Z_x} = \frac{1}{2}\frac{Z_x^2 - R_o^2}{Z_x R_o}$$

$$S = (-1/2)R_o/Z_x \qquad \text{for } |Z_x| \ll R_o$$

$$= 0 \qquad \text{for } |Z_x| = R_o$$

$$= (1/2)Z_x/R_o \qquad \text{for } |Z_x| \gg R_o.$$

This implies that this type of transducer has little sensitivity to the voltmeter gain variance when the DUT impedance is near R_o (50 ohms). The gain variance of the voltmeters behaves as an offset impedance with a magnitude of $(1/2)R_o |\Delta\alpha_r/\alpha_r|$ when the DUT impedance is much smaller than R_o, where $\Delta\alpha_r$ is the change in the gain ratio α_r. The gain variance of the voltmeters behaves as an offset admittance with a magnitude of $(1/2)G_o|\Delta\alpha_r/\alpha_r|$ when the DUT impedance is much larger than R_o, where $G_o = 1/R_o$.

Fig. 4 shows this characteristic.

I-V Method. Fig. 5 shows the simplest transducer for the I-V method. The bilinear transformation in this case is:

$$Z_x = R_o V_r, \tag{4}$$

where $R_o = 50$ ohms is a resistor that converts the DUT current to a voltage and $V_r = V_2/V_1$. The parameters in equation 1 are $K_1 = R_o$, $K_2 = 0$, and $K_3 = 0$.

We also assume that there is some gain variance in the vector voltmeters in Fig. 5. The measured voltages V_1 and V_2 are related to the calculated impedance Z_x as follows:

$$V_1 = \alpha_1 E \frac{R_o}{Z_x + R_o}$$

$$V_2 = \alpha_2 E \frac{Z_x}{Z_x + R_o}$$

where α_1 is the gain of vector voltmeter 1, α_2 is the gain of vector voltmeter 2, and $V_r = V_2/V_1$ is the voltage ratio. Thus,

$$Z_x = R_o V_r\, \alpha_r,$$

where $\alpha_r = \alpha_1/\alpha_2$ is the ratio of the voltmeter gains.

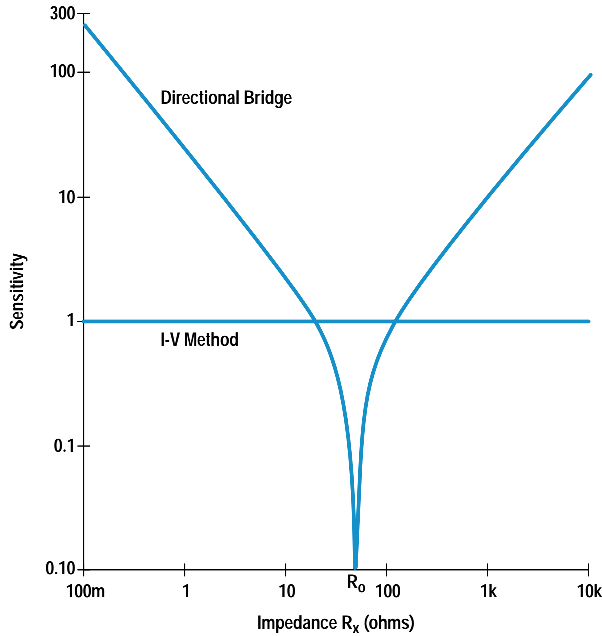


Fig. 4. Sensitivity of the directional bridge and I-V methods to voltmeter gain variance.

In this case, the sensitivity is given by:

$$S = \frac{\delta Z_x / Z_x}{\delta \alpha_r / \alpha_r} = 1.$$

The error ratio $(\Delta Z_x/Z_x)/(\Delta\alpha_r/\alpha_r)$ is always constant and equal to unity. For example, if the voltmeter gain ratio α_r changes by 1%, an impedance error of 1% is incurred for any DUT.

New RF Impedance Analyzer

The foregoing analysis shows that the voltmeter gain variance is neither suppressed nor magnified for all DUT impedances by an I-V method transducer. This characteristic is desirable for wide impedance measuring capability. Therefore, we adopted this type of transducer for the new HP 4291A one-port RF impedance analyzer.

Fig. 6 shows the basic circuit of the transducer in the HP 4291A, which is a modified version of Fig. 5. The high-impedance configuration (switch closed) realizes perfect open and imperfect short conditions, while the low-impedance configuration (switch open) realizes imperfect open and perfect short conditions.* In either switch position, the output impedance at the DUT port is always R_o.

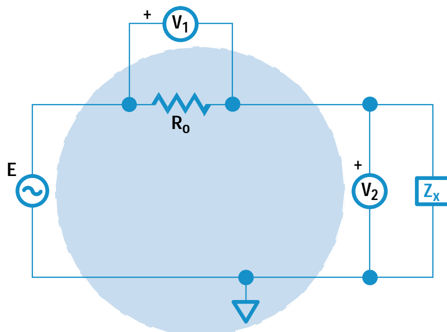


Fig. 5. I-V method.

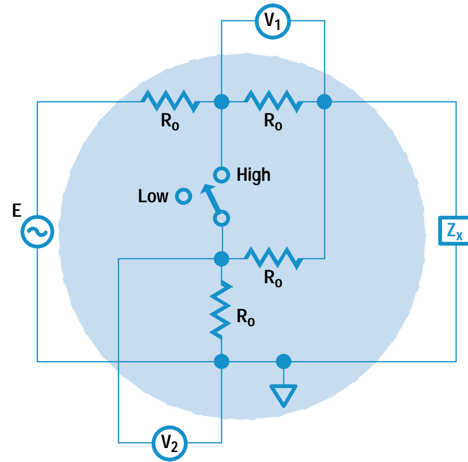


Fig. 6. Basic transducer circuit of the HP 4291A RF impedance analyzer with switch indicating circuit configurations for high-impedance and low-impedance measurements.

Fig. 7 shows the actual HP 4291A transducer circuit configurations. A number of considerations influenced the design of these circuits. First, because a wideband switch with small nonlinearity and small transients over a wide signal range is not easily realized, we divided the circuit of Fig. 6 into two separate circuits. Second, because the minimum frequency of the analyzer is 1 MHz, the floating voltmeter (V_1), which corresponds to the current meter, is easily realized by using a balun. Third, we adopted a circuit in which the voltmeter readings change by the same percentage if the floating impedance of the balun changes. Thus, the voltage ratio V_r does not change and stable impedance measurements are realized.

Block Diagram

A conceptual block diagram of the HP 4291A including the transducer discussed above is shown in Fig. 8. Only the parts relevant to this discussion are included. The mainframe is similar to the HP 4396A network and spectrum analyzer.[3]

Two key features of the HP 4291A are time division multiplex operation and impedance ranging. Two voltmeters are obtained by time division multiplexing one voltmeter. The multiplexing period is 2 ms. This ensures that slow drift of the voltmeter gain does not affect the impedance measurement. With this method, the signal path after the multiplexer can be extended. The HP 4291A uses a 1.8-m cable between the transducer and the instrument mainframe. This allows wide flexibility in constructing a test system using automatic device handlers. The single-path configuration results in good temperature characteristics even with an extended cable.

At frequencies below 200 MHz there is an expanded range. In the expanded range there is a gain difference between the voltage channel and the current channel ahead of the multiplexer. This impedance ranging offers stable measurements for DUTs with impedances that differ greatly from 50 ohms.

Fig. 9 shows HP 4291A impedance measurement specifications. General error factors are the uncertainties of standards

* A perfect open condition means that voltmeter V_1 reads zero with the DUT port open. A perfect short condition means that voltmeter V_2 reads zero with the DUT port shorted.

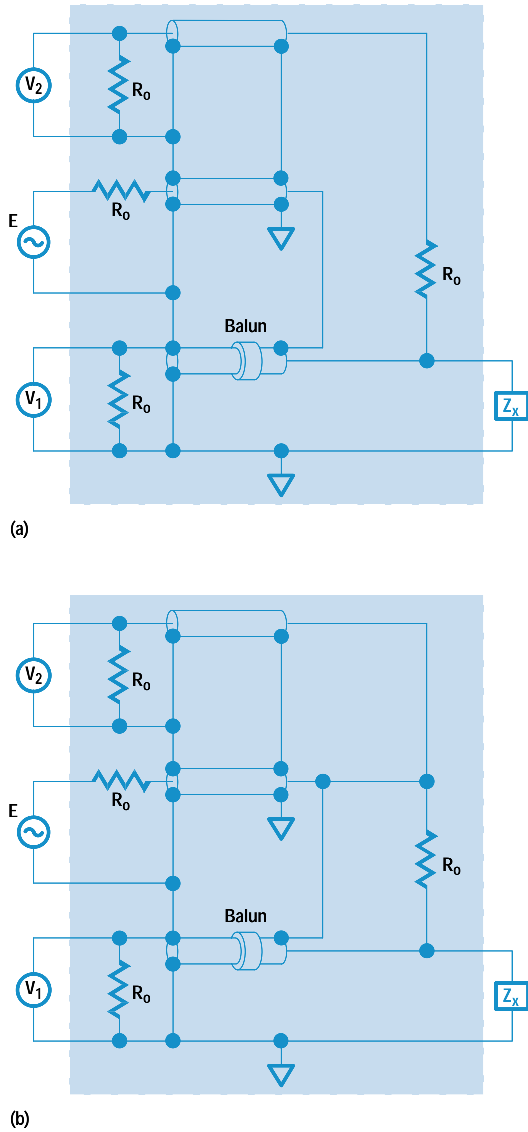


Fig. 7. (a) Actual HP 4291A transducer circuit for low-impedance measurements. (b) Circuit for high-impedance measurements.

used in the calibration, instabilities, and interpolation errors. The instabilities consist mainly of connection nonrepeatability, long-term drift, circuit nonlinearity, temperature coefficients, and noise. The instabilities and interpolation errors are small enough that the impedance phase errors can be reduced by means of the special calibration discussed next.

High-Q Measurements

Normally, the accuracy requirement for impedance phase measurements is greater than that for impedance magnitude measurements. The HP 4291A has a special, easy-to-use, calibration for measurements of devices having high Q (quality factor).

Even if the stability of the instrument is good enough, accurate Q measurements cannot be made without adequate phase calibration. For instance, if we want to measure the Q factor with 10% uncertainty for a DUT whose Q value is 100, the uncertainty in phase must be smaller than 10^{-3}. The phase accuracy of the instrument is determined almost entirely by the uncertainty of the 50-ohm load standard used in the OSL calibration. One way to improve phase measurement accuracy is to use a phase-calibrated load standard. However, it is not guaranteed that the phase uncertainty for a calibrated 50-ohm load is smaller than 10^{-3} at high frequencies, such as 1 GHz.

Another way to improve phase measurement accuracy is to use, in addition to the normal open-short-load standards, a low-loss air capacitor as a second load (LOAD2). The dissipation factor (D) of the air capacitor should be below 10^{-3} at around 1 GHz. With this method, the uncertainty in the measured phase is decreased from the phase uncertainty of the 50-ohm load (LOAD1) to the uncertainty of the dissipation factor D of the low-loss capacitor (LOAD2) for almost all DUT impedances. The next section gives the details of this method.

Modified OSL Calibration[2]

We want a calibration method that reduces the error in phase measurement in spite of the existence of phase error for the 50-ohm load. We have the 50-ohm load standard whose impedance magnitude is known but whose impedance phase is not. We add another load (LOAD2) whose impedance phase is known but whose impedance magnitude is not. We use a low-loss capacitor as the second load. There are still at most three unknown circuit parameters. However, two more unknowns related to standards are added. Let us define the problem. There are eight real unknown parameters:

- Circuit parameter K_1 (two real parameters)
- Circuit parameter K_2 (two real parameters)
- Circuit parameter K_3 (two real parameters)
- The impedance phase θ_{ls1} of the 50-ohm load (one real parameter)
- The impedance magnitude Z_{abs_ls2} of the low-loss capacitor LOAD2 (one real parameter).

We have solved this problem analytically. For the simplest case where both the open and the short standard are ideal, the three circuit parameters are found as follows:

$$K_1 = AZ_{lsi}R_o$$
$$K_2 = -Z_{sm}/R_o \qquad (5)$$
$$K_3 = -Y_{om}R_o$$

where:

R_o = characteristic impedance
$A = (1 - Z_{lmi}Y_{om})/(Z_{lmi} - Z_{sm})$
Y_{om} = measured admittance for open standard
Z_{sm} = measured impedance for short standard
Z_{lmi} = measured impedance for load standard i (i = 1 for LOAD1, i = 2 for LOAD2)
Z_{lsi} = true impedance for load standard i
$Z_{ls1} = Z_{abs_ls1}\exp(j\theta_{ls1})$
$Z_{ls2} = Z_{abs_ls2}\exp(j\theta_{ls2})$
$\theta_{ls1} = \theta_2 - \theta_1 + \theta_{ls2}$
$Z_{abs_ls2} = (A_1/A_2)Z_{abs_ls1}$
Z_{abs_ls1} = impedance magnitude for LOAD1 (50-ohm, known)
θ_{ls2} = impedance phase for LOAD2 (low-loss capacitor, known)
$\theta_1 = \arg((1 - Z_{lm1}Y_{om})/(Z_{lm1} - Z_{sm}))$
$\theta_2 = \arg((1 - Z_{lm2}Y_{om})/(Z_{lm2} - Z_{sm}))$

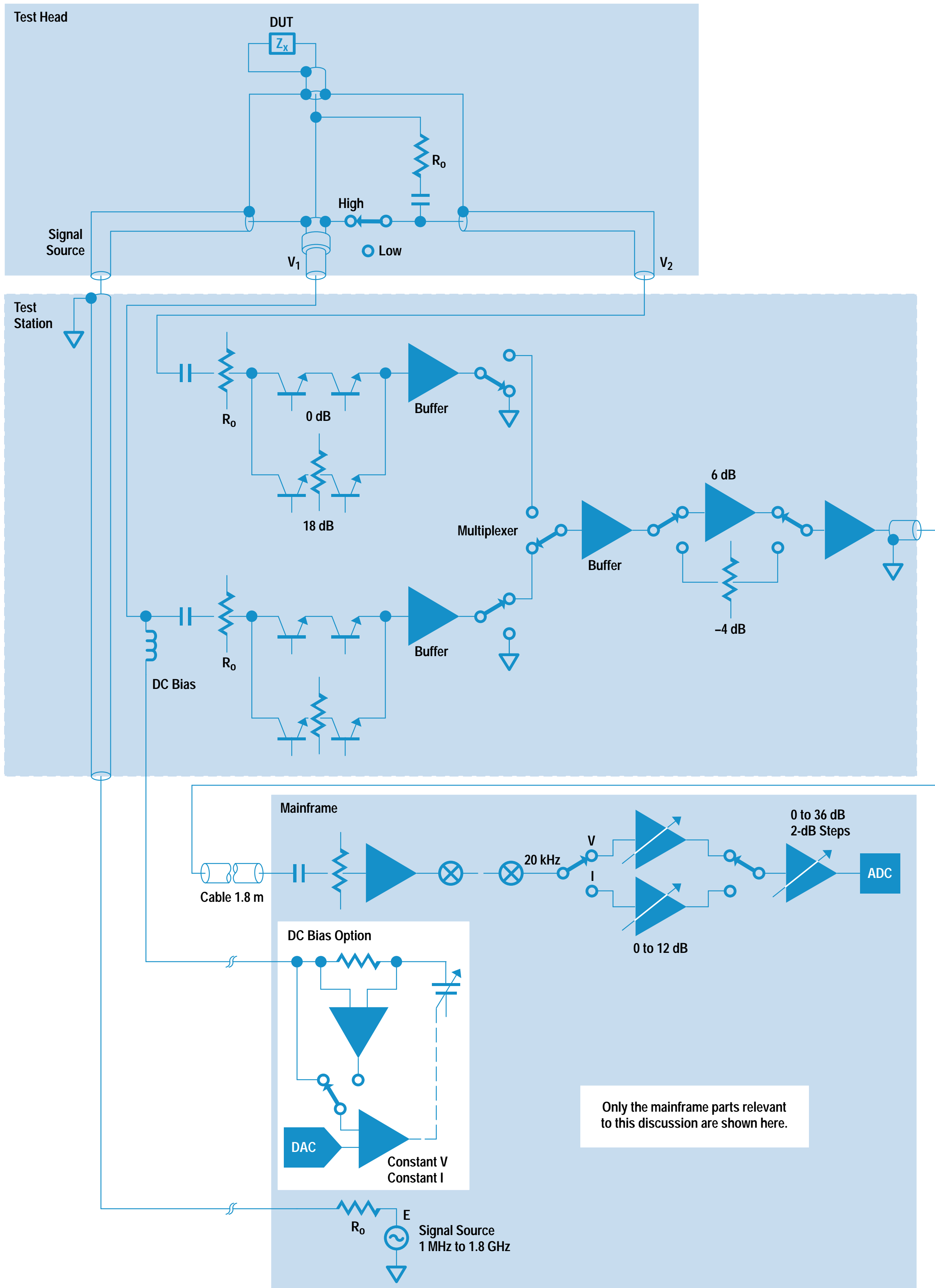


Fig. 8. Conceptual block diagram of the HP 4291A with switch (in test head) showing the difference between the transducers for high-impedance and low-impedance measurements.

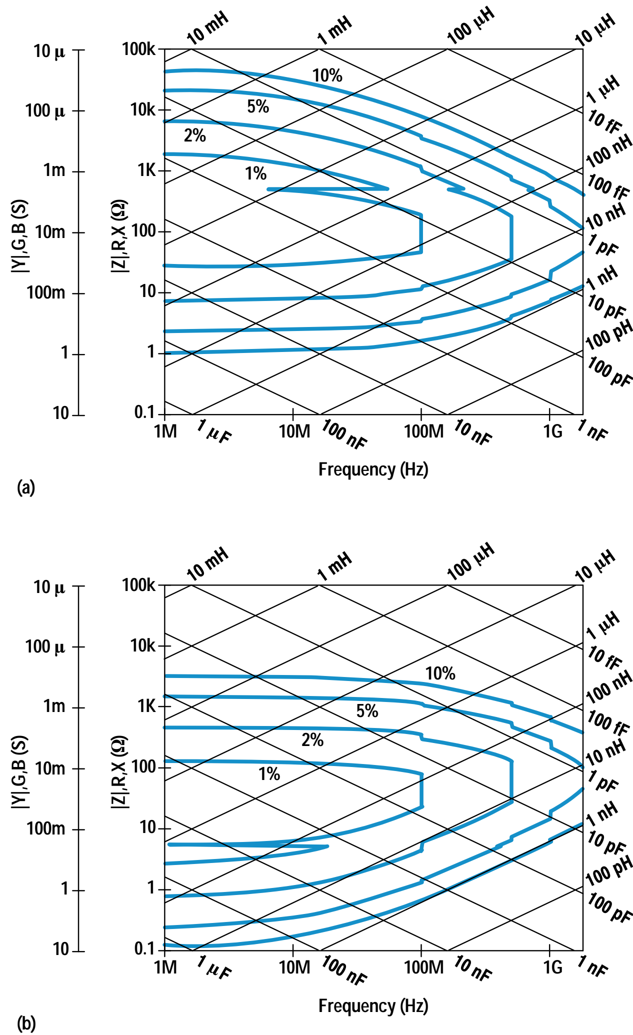


Fig. 9. (a) Errors for impedance magnitude with the transducer for high impedance. (b) Errors for impedance magnitude with the transducer for low impedance.

$$A_1 = |(1 - Z_{lm1}Y_{om})/(Z_{lm1} - Z_{sm})|$$
$$A_2 = |(1 - Z_{lm2}Y_{om})/(Z_{lm2} - Z_{sm})|$$

For the actual case these circuit parameters are expressed by far more complicated equations. Therefore, we adopted a simpler procedure consisting of two steps. Step 1 is as follows:

- Regard the impedance of the 50-ohm load as $Z_{ls1} = 50+j0$ (that is, the phase of LOAD1 is zero).
- Find the circuit parameters K_1, K_2, and K_3 by normal OSL calibration using the load value Z_{ls1}.
- Execute correction for LOAD2 and get the corrected impedance Z_{corr2}.
- Calculate the phase difference $\Delta\theta$ between the phase of Z_{corr2} and the true phase of LOAD2.

Step 2 is as follows:

- Modify the impedance of LOAD1 to Z'_{ls1} whose phase is $-\Delta\theta$ and whose impedance magnitude is still 50 ohms.
- Calculate the circuit parameters again by normal OSL calibration using the modified load impedance Z'_{ls1}.

Although this is an approximate method, it is accurate enough for our purposes. We call this method the modified OSL calibration.

Phase Measurement Errors

The following error factors affect phase measurement accuracy using the modified OSL calibration:

- Uncertainty in the impedance magnitude of LOAD1
- Impedance phase of LOAD1
- Impedance magnitude of LOAD2
- Uncertainty in the impedance phase of LOAD2
- Uncertainty in the admittance magnitude of the open standard.

Notice that the second and third factors would not cause any error if we were using the analytical solution. Computer simulations have shown that:

- The phase measurement error caused by the uncertainty in the impedance magnitude of LOAD1 is small.
- The phase measurement error caused by the impedance phase of LOAD1 is small.
- The phase measurement error caused by the impedance magnitude of LOAD2 is small.
- The uncertainty in the impedance phase of LOAD2 directly affects the phase measurement error.
- For reactive DUTs, the phase measurement error caused by the uncertainty in the admittance magnitude of the open standard ($|\Delta Y_{open}|$) is reduced to $|R_o\Delta Y_{open}|(C_{open}/C_{ls2})$ in the modified OSL calibration, where R_o is the characteristic impedance (50 ohms), C_{open} is the capacitance of the open standard, and C_{ls2} is the capacitance of LOAD2 (low-loss capacitor). In the case of resistive DUTs, the error is the same as in the normal OSL calibration.

Fig. 10 shows the relationship between the phase error $|\Delta\theta|$ and the DUT impedance when the LOAD2 phase uncertainty $|\Delta\theta_{ls2}|$ is 500×10^{-6} radian in the modified OSL calibration. The relationship between $|\Delta\theta|$ and the DUT impedance is shown in Fig. 11 when the open admittance uncertainty $|\Delta Y_{open}|$ is 5 fF in the modified OSL calibration.

In summary, the phase measurement error when using the modified OSL calibration is mainly determined by the uncertainty in the impedance phase of LOAD2 and the uncertainty in the admittance magnitude of the open standard. We now evaluate these two items. The D factor for the capacitor (3 pF) used in the calibration can be small because the capacitor's dimensions are small and the space between the inner and outer conductors is filled almost completely with air.

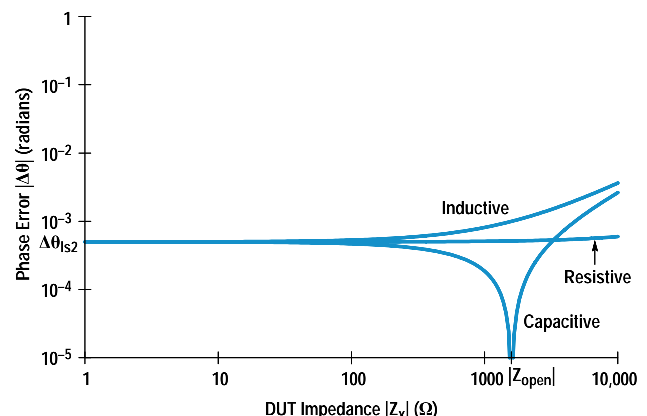


Fig. 10. Relationship between phase measurement error and the uncertainty $\Delta\theta_{ls2}$ in the impedance phase of LOAD2 (low-loss capacitor) at 1 GHz for resistive and reactive DUTs. $|Z_{open}|$ is the impedance of the open standard (about 2000 ohms).

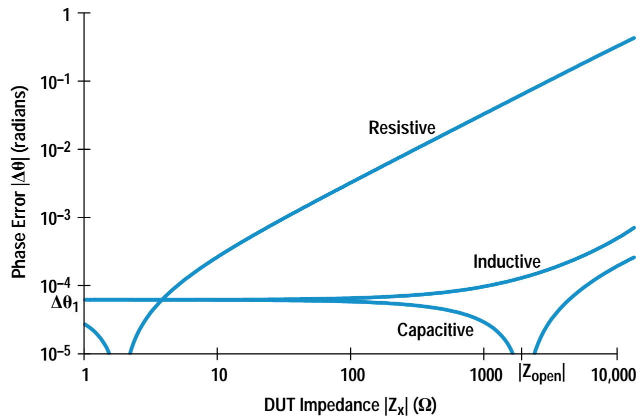


Fig. 11. Relationship between phase measurement error and the uncertainty in the admittance magnitude of the open standard at 1 GHz for resistive and reactive DUTs. $\Delta\theta_1 = |R_o \Delta Y_{open}| C_{open}/C_{ls2}$ where R_o is the characteristic impedance (approximately 50 ohms), $|\Delta Y_{open}|$ is the uncertainty in the admittance magnitude of the open standard (about 30 μS), C_{open} = 100 fF, and C_{ls2} = 3 pF. $|Z_{open}|$ is the impedance of the open standard (about 2000 ohms).

The D value has been estimated as 500×10^{-6} at 1 GHz in a residual resistance measurement at the series resonant frequency. The D factor increases with frequency f as $f^{1.5}$ because of the skin effect. By using zero as the D value for the capacitor during calibration, a phase measurement error of 500×10^{-6} is incurred at 1 GHz. The uncertainty for the open capacitance is ±5 fF at most, leading to a phase measurement error less than $\pm100\times10^{-6}$ at 1 GHz. Overall, a phase measurement uncertainty of 500×10^{-6} is incurred by using the modified OSL calibration.

Impedance Traceability

For an impedance performance check using the top-down method[4] we set up a kit traceable to U.S. national standards. This kit is calibrated annually at our in-house standards laboratory. Two major items in the kit are the 50-ohm load and a 10-cm-long, 50-ohm, beadless air line. The 50-ohm load is desirable because its frequency characteristic for impedance is very flat. The structure of the air line is very simple, so it is easy to predict its frequency characteristic and it is convenient to realize various impedances by changing frequencies with the line terminated in an open or short circuit.

The traceability path for the kit is shown in Fig. 12. The impedance characteristic of open-ended and short-ended air lines can be calculated theoretically from their dimensions and resistivity.[5] However, it is not easy to design a system to calibrate the dimensions of the air line in each individual kit. Therefore, only the dimensions of the reference air line of our standards laboratory is periodically calibrated. Calibration of the individual air line is executed by a network analyzer calibrated from the reference air line. The 50-ohm load calibration is done mainly by the quarter-wave impedance method and dc resistance measurement. The open termination is calibrated by a capacitance bridge at low frequencies and by the network analyzer at high frequencies. The short termination is treated as ideal. Uncertainties for the short termination consist of skin effect and nonrepeatabilities.

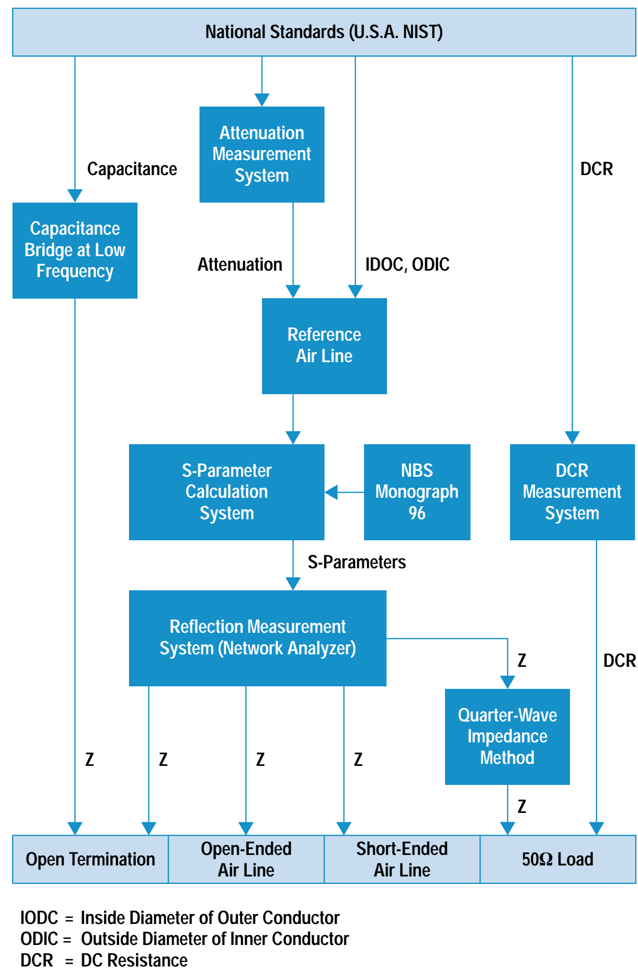


Fig. 12. Traceability path for the performance test kit.

Test Fixtures

In actual measurements, test fixtures are needed to accommodate different-shaped DUTs. As the frequency range goes up, fixtures that are able to handle smaller devices are needed. We have developed four types of fixtures:

- A fixture for surface mount devices with bottom electrodes
- A fixture for surface mount devices with side electrodes
- A fixture for very small surface mount devices
- A fixture for leaded components.

To reduce the error at the fixture terminal it is necessary to minimize the length from the reference plane of the APC-7 connector to the fixture terminal and to minimize the connection nonrepeatability. The new fixtures' repeatability is almost five times better than our old ones. The typical nonrepeatability of the surface mount device fixtures is ±50 pH and ±30 mohms for short-circuit measurements and ±5 fF and ±2 mS for open-circuit measurements.

Fixture compensation is included in the firmware corresponding to correction at the fixture terminal. This reduces the errors generated in the circuit between the reference plane and the fixture terminal. The best compensation method is the OSL method. However, it is not easy to prepare a standard load having excellent frequency characteristics. As

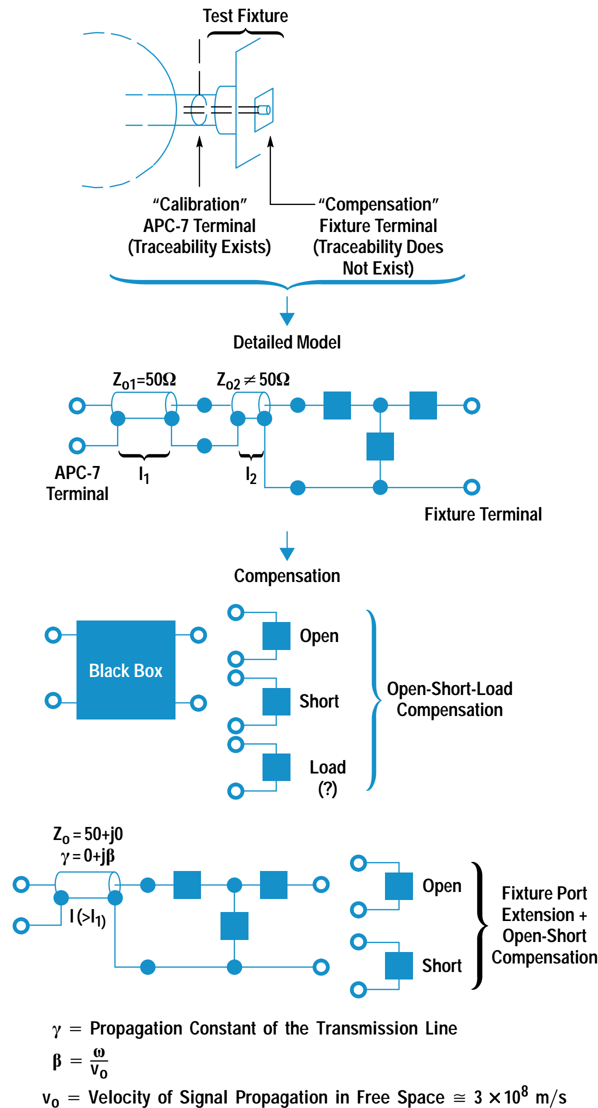


Fig. 13. Open-short-load (OSL) fixture compensation versus HP 4291A fixture port extension plus open-short compensation.

a more realistic alternative we provide another compensation function: fixture port extension combined with open-short correction at the fixture plane. This method assumes that there is a short transmission line between the APC-7 terminal of the transducer head and the DUT connector of the fixture (see Fig. 13). When the user selects one of our new fixtures on the HP 4291A display, an appropriate fixture port extension value—an equivalent length of ideal 50-ohm line previously determined for that fixture—is automatically set. The user then performs a compensation using open and short circuits.

The difference between the OSL method and the HP 4291A method is that the OSL method assumes ideal open, short, and load standards while the HP 4291A method assumes ideal open and short standards and an ideal transmission line. We feel that the assumption of an ideal transmission line is more realistic than the assumption of an ideal load for compensation over a wide frequency range.

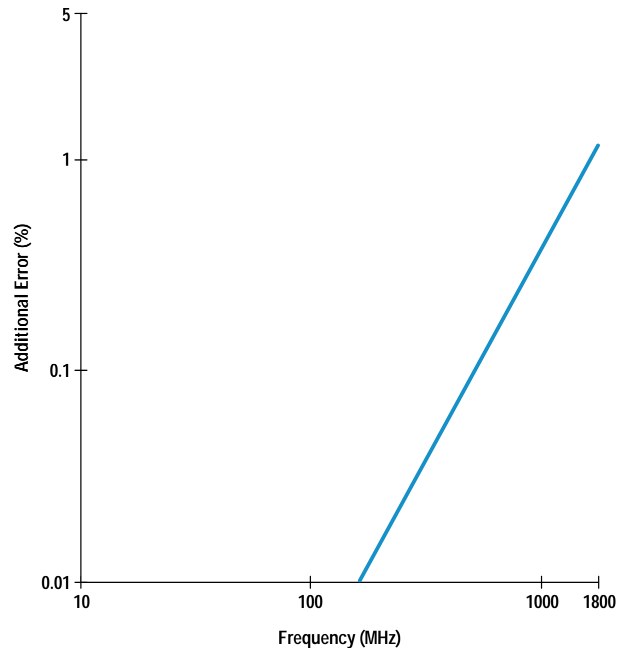


Fig. 14. Typical error contributions of the new test fixtures after fixture compensation.

Fig. 14 shows the typical test fixture error contributions when using this compensation function. These values are almost three times better than the errors for our former type of fixtures.

Conclusion

Selection of a transducer (as defined on page 68) is important for accurate impedance measurement. A new type of transducer based on the current-voltage method and having wide impedance measuring capability is used in the new HP 4291A RF impedance analyzer. A new phase calibration technique, a modified OSL calibration, has also been developed. It uses a low-loss capacitor as the second load and makes accurate Q measurements possible.

Acknowledgments

The author is grateful to K. Yagi, S. Tanimoto, and H. Wakamatsu, who provided helpful discussions.

References

1. T. Yonekura and M. Jansons, "High-Frequency Impedance Analyzer," *Proceedings of CARTS—Europe '93,* pp. 44-52.
2. T. Yonekura, "A New Calibration Method for Impedance Meters," *Proceedings of the 1994 IEEE Instrumentation and Measurement Technology Conference,* Shizuoka, Japan, May 1994, pp. 1004-1007.
3. S. Kawabata and A. Nukiyama, "A High-Performance 1.8-GHz Vector Network and Spectrum Analyzer," *Hewlett-Packard Journal,* Vol. 44, no. 5, October 1993, pp. 76-84.
4. *Traceability and the HP 8510 Network Analyzer,* Hewlett-Packard publication no. 5954-1552, November 1, 1985.
5. R.E. Nelson and M.R. Coryell, *Electrical Parameters of Precision, Coaxial, Air-Dielectric Transmission Lines,* NBS Monograph 96, June 30, 1966.

Virtual Remote: The Centralized Expert

Remote operation of bit error rate test sets using an X Windows based "virtual instrument" allows network operators to monitor remote sites from a central office. The extensive use of a common firmware development platform allowed the fast-track development of virtual remote software and rapid integration into all instruments built using the platform.

by Hamish Butler

In today's competitive marketplace, network operators must provide their customers with a cost-effective and efficient service. Large corporate customers expect the network operator to provide a guaranteed quality of service at the best possible price. This means that the network operator has to run the network as efficiently as possible and as cheaply as possible. Let's look at the issues of efficiency and cost separately.

First, the network must be efficient. That is, it must reliably transport customer data on demand. It must have a low error rate and high uptime. If data is lost in transit or if customers are unable to send data because the network is down they are likely to switch to another carrier. It is no longer acceptable for the network operator to wait for faults to be reported and then to fix them; a preventive maintenance approach is required. Preventive maintenance involves monitoring the performance of the network links and looking for any degradation in performance. If the performance degrades beyond a certain level the customer traffic must be switched to another data link with the required level of performance. The defective link must then be fixed as quickly as possible.

Network operators must be able to provide a competitively priced service to their customers and still be able to run the network and make a profit. This places heavy demands on network test and maintenance:

- Test and maintenance departments are under pressure to maintain the required level of service using fewer staff.
- Expenses are under tight control. Traveling between sites is expensive and unproductive.
- Network capacity is at a premium. The amount of spare capacity that can be held in reserve is limited. If a data link has errors and traffic is switched to a spare link it is probable that the errored link must be fixed as soon as possible so that it can then provide the spare capacity.

Traditional approaches have used comparatively low-skilled technicians to provide the first level of troubleshooting and maintenance. These technicians may have been based at remote sites or based at a central site and dispatched to remote sites as required. If these technicians were unable to diagnose a fault a more skilled technician was sent to the remote site. This process is time-consuming, expensive, and prone to error. It is not unknown for a technician to travel to a remote site only to find that the fault lies somewhere else. Sometimes the unskilled technician has even introduced faults into the system while attempting to diagnose the original fault.

The ability to perform testing of remote sites from a central office has many advantages. Skilled technicians can be concentrated at one site. Less time is spent traveling. Fault locations can be diagnosed before dispatching technicians to fix the faults. Preventive maintenance is improved through the ability to monitor many remote sites from one place.

The traditional approach to such centralized testing has been to locate portable test equipment at remote sites and to communicate with these instruments using an RS-232 link and modems. Using this method it was normal to interrogate the instrument by writing a controller program for a PC or workstation. This method is restricted because of the limited amount of information that is returned to the operator. The operator is only given the information that the programmer requested when the program was written. For example, if the program is monitoring one or more selected results and something that is not being monitored by the program changes, there will be no immediate feedback.

It is sometimes necessary for the skilled technician at the central site to give instructions to an unskilled technician at the remote site. Using the above method the skilled technician has no direct feedback of the tasks performed by the unskilled technician.

The Virtual Instrument

Engineers at the HP Queensferry Telecommunications Operation have been developing instrument firmware using an X Windows instrument simulation for several years. The engineers responsible for the instrument simulator had often thought how useful it would be if the simulator could be used to control an actual instrument.

In the second half of 1991 an HP field engineer started working with a large network operator on a contract for HP telecommunications test sets and computers. This customer was setting up a large-scale centralized test and maintenance system to monitor the network. Working with the customer the HP field engineer developed the idea of the virtual instrument.

Fig. 1. An image of two instruments in a "dark office" superimposed on an image of an operator at a workstation using virtual remote software to control the two instruments. Representations of the instruments' front panel are displayed on the screen.

This was the vision presented to the customer: From a single central office, customer personnel would be able to use an HP workstation to bring up an accurate simulation of a remote instrument on the display. They would be able to display several instruments simultaneously. Fig. 1 illustrates this concept. The photograph shows an operator in a central office using virtual remote to control two instruments in a distant "dark office." Each virtual instrument would be operated by using the mouse to press keys. These key presses would be relayed to the remote instrument. As the remote instrument updated any of its feedback mechanisms—display, LEDs, or audio—the instrument simulation would relay these changes to the operator in the central office.

The important task now was to see how R&D could best implement the virtual instrument application. The development of the product was split into two stages. The first was to take the current instrument simulator and use it to produce a prototype virtual instrument application. The second stage was to take the prototype application and turn it into a polished software product that would meet the rigorous demands of HP customers.

Product Description

The product that emerged from this effort is the HP 15800A virtual remote capability software. It runs on HP 9000 Series 300, 400, or 700 workstations under the HP-UX* 8.0 operating system.† The software provides for the centralized supervision, operation, and collection of results from remotely located HP 377xxA Option V01 telecommunications test sets and analyzers. Option V01 virtual remote capability enhances the instrument firmware to respond to the HP 15800A software.

A single workstation can control up to twelve remote test sets. The display shows windows identical to the front panels and screens of the test sets. When a technician is controlling a remote test set manually, all actions and results can be monitored at the workstation.

At present, Option V01 is available for the following test sets:

- HP 37701B T1 tester
- HP 37702A digital data tester
- HP 37704A SONET test set
- HP 37714A PDH/SDH test set
- HP 37717A PDH/SDH test set
- HP 37721A digital transmission analyzer
- HP 37722A digital telecomm analyzer
- HP 37724A PDH/SDH test set
- HP 37732A telecomm/datacomm analyzer.

Option V01 can be retrofitted to existing instruments by changing ROMs.

Virtual Remote Design Concept

As has already been explained, instruments at the Queensferry Telecommunications Operation are developed using an instrument simulator. This instrument simulator is in fact part of a larger common firmware platform.[1] This common firmware platform incorporates compiler-based code generation and simulation tools along with the source code for a core generic instrument. The design of virtual remote is so closely tied to the instrument simulator that a brief explanation is required.

The core of Queensferry Telecommunications Operation's common firmware platform is a compilation and simulation tool known as ISS—Instrument Software System. ISS is based on an abstract, high-level, instrument definition language. This language is used to define many aspects of the instrument operation:

- The instrument user interface
- All instrument control variables
- Interaction and dependencies between control variables
- Instrument display data—text and graphics
- Instrument results
- Instrument data input—hard and soft keys
- Relationships between data input (keys), control variables, and data output (display data)
- Printer output data
- Remote control command definitions.

The instrument firmware development process is as follows:

- Produce the instrument definition using the ISS language
- Simulate the instrument operation using the ISS simulator
- Use the ISS compiler to generate an embedded instrument database
- Compile the instrument database into embedded instrument code
- Test firmware operation in the target instrument.

The ISS simulator and the target instrument share the core components, mainly the ISS database processing engine. The development process is illustrated in Fig. 2.

As shown in Fig. 2, the ISS database and the ISS engine are common to both the ISS simulator and the actual instrument. The instrument operator interacts with the instrument using keys on its front panel and data displayed on its screen. On the workstation, the key input and screen output are handled by a set of functions written in the C programming language. These functions are implemented using X Windows function calls.

In the instrument the screen output and key input are handled by an alternative implementation of the same C functions. In this case the functions interact directly with the instrument's display control and keyboard input hardware.

The basic instrument architecture is shown in Fig. 3.

† A PC version, HP 15801A, is now available as well.

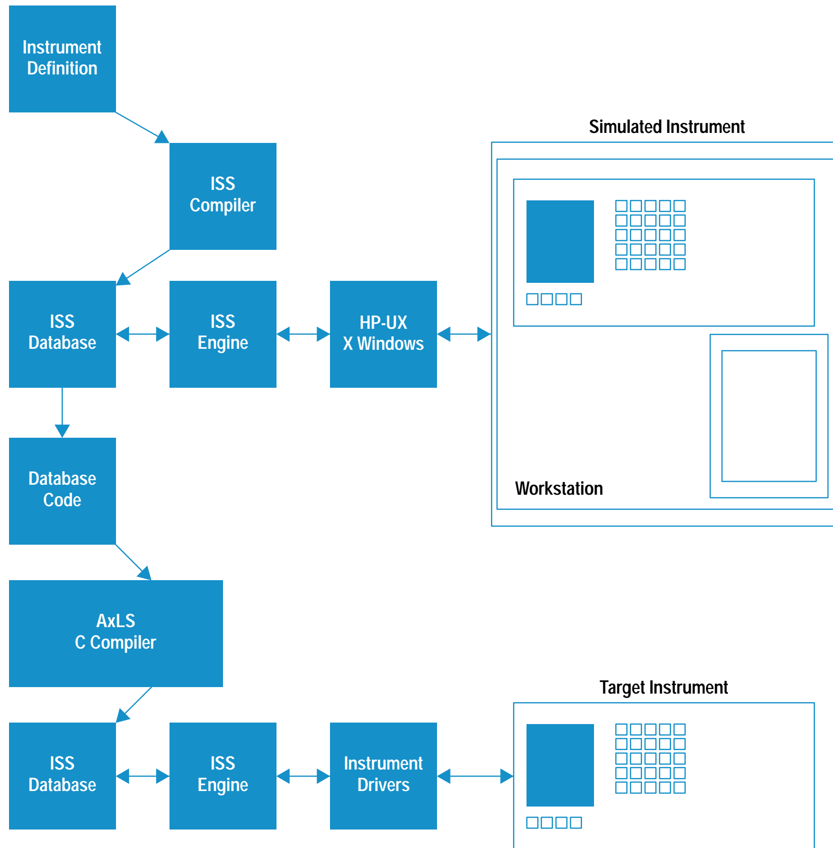


Fig. 2. Instrument development using the ISS compiler and simulator. The ISS database and the ISS engine are common to both the simulator and the actual instrument.

The key to the development of virtual remote was to connect the ISS engine embedded in the instrument with the screen output and key input functions used on the workstation. The use of the same set of display functions in both the workstation and the embedded system gave the potential to update both displays at the same time. We also wanted to duplicate the function calls made by the instrument. At the same time as the function call is performed in the instrument it must also be performed on the workstation. This technique is known as a remote procedure call. A UNIX® application

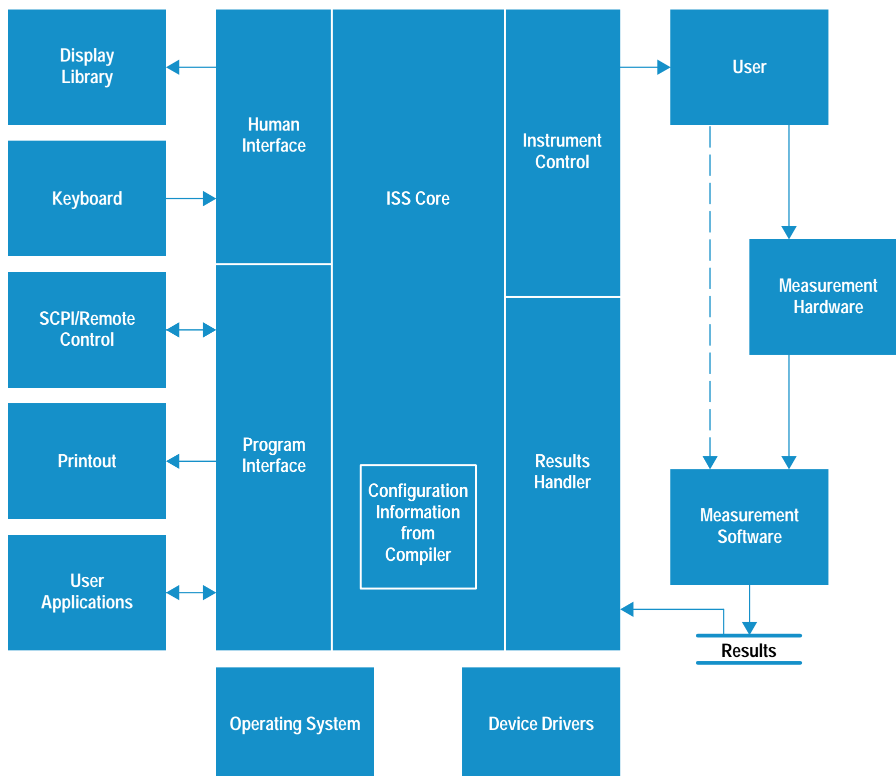


Fig. 3. Basic instrument architecture.

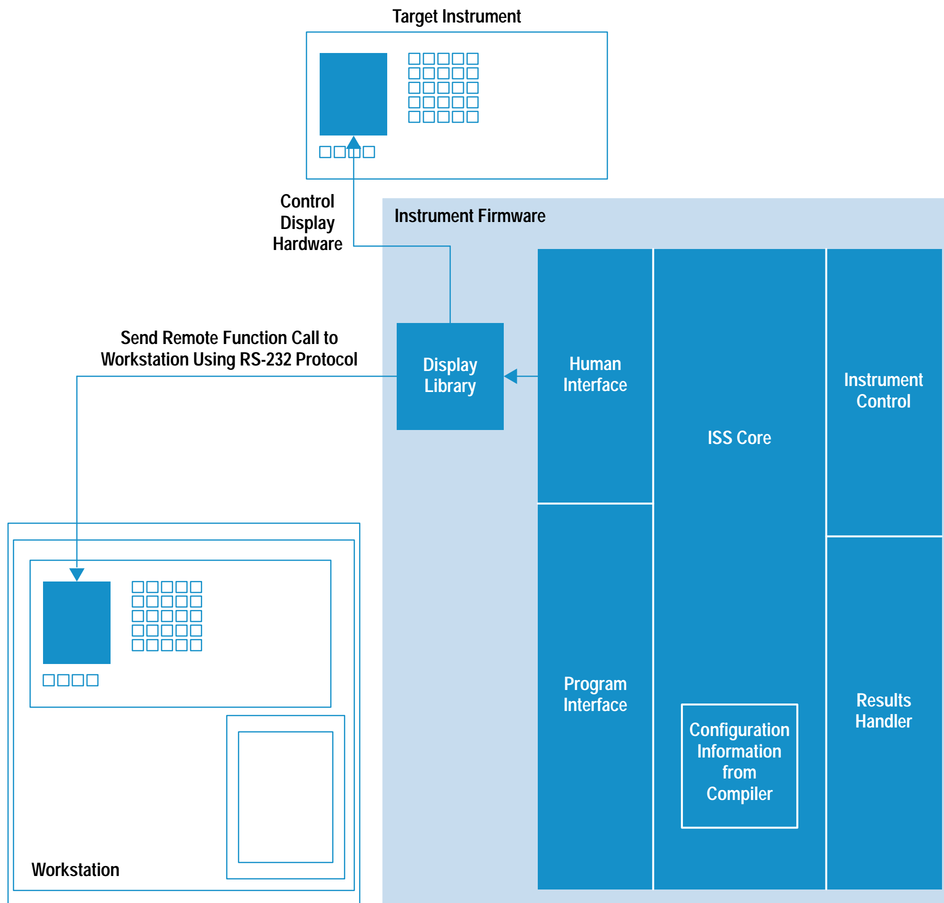


Fig. 4. Instrument-to-virtual-display connection.

that needs to make remote procedure calls to another machine on the network would use Sun Microsystems' RPC. In our instrument environment we could not support this. It was decided to develop a protocol of our own between the instrument and the workstation that could be used to initiate remote display functions. This protocol would use ASCII data transmitted between the instrument and the workstation on a 9600-baud RS-232 data link. This is illustrated in Fig. 4.

The ISS database engine is event-driven. The keyboard process in the instrument waits for a key to be pressed and then acts upon that key. In the case of the display we had to output to two devices at once—the simulated instrument and the target instrument. In the case of the keyboard we had to accept input data from two sources—a local keyboard and a remote keyboard. This is accomplished through the use of our dedicated protocol. Keys pressed locally are processed as they always were. A key pressed on the workstation is encoded and transmitted over the RS-232 link to the instrument. When the data is received by the instrument it is decoded and the key code is placed in the key input queue. This is shown in Fig. 5.

It would have been possible to use the above techniques to develop an application that ran on the workstation and was used to communicate with a remote instrument. This application would have to have embedded knowledge about the instrument it communicated with so that it could render an accurate representation of the instrument on the workstation screen and have knowledge of the keys present on the instrument front panel and the key codes assigned to each one.

Because our division manufactures many different instruments, this solution would require a separate virtual remote application for each instrument, which is unsatisfactory for the following reasons:

- Developing multiple applications would require extra engineering effort.
- Extra administration effort would be required for sales and support.
- Customers often use several different instruments. Some applications require different instruments used in combination. It would be unfriendly and expensive for our customers to have to buy and use a separate application program for each different instrument.
- This approach may have presented problems with new instruments or instrument enhancements released after the application code was produced.

The ideal solution is to have a single virtual remote application program that can be used with any instrument that is already shipping or that may ship in the future. To do this the virtual remote application has to be completely generic, sourcing all of its instrument-specific data from the instrument itself.

Once again, the use of the ISS simulator for instrument development helped us to find the solution we required. The ISS simulator already displayed a representation of the instrument on the workstation and had knowledge of the instrument key positions and key codes. This data is contained in the instrument definition coded using the ISS language. In the simulator this data is compiled into RAM data structures

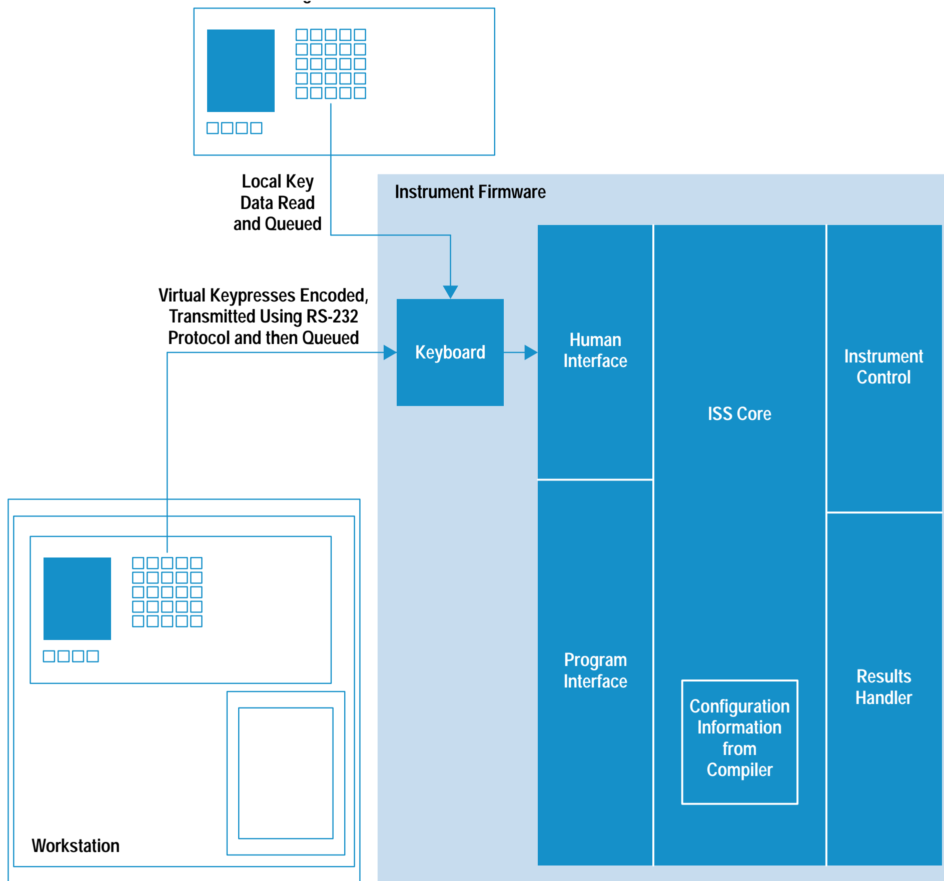


Fig. 5. Virtual-keyboard-to-instrument-keyboard process.

which are then used to render the instrument and process the keys.

The development of virtual remote required these data structures to be added to the instrument database embedded into the instrument. Once this was done it was an easy step to enhance the instrument-to-workstation protocol to allow the data held in ROM in the instrument to be sent over the RS-232 data link to the virtual remote application in the workstation. The virtual remote application then holds the data in a RAM data structure and uses it to render the instrument and process key presses.

Fig. 6 shows a workstation screen displaying the virtual remote representation of an instrument being controlled. All of the data needed to draw this image is stored within the instrument and is sent to the virtual remote application upon request. This allows the virtual remote software to control many different instruments without any prior knowledge of how they look or how they operate.

Virtual Remote Development

Using the above design concepts it was relatively easy to produce a working virtual remote prototype. The prototype application was extremely useful and very successful. It proved that the design concept was feasible, it illustrated the concept of the virtual instrument, it allowed the instrument-to-workstation protocol to be tested, it helped prove the viability of the product, and it gave a base upon which the final product would be built.

Once the prototype was complete the next stage of the project was to take the prototype and turn it into an application suitable for use by customers. This application had to implement features not supported in the initial prototype:

- Multiple simultaneous connections
- Connection arbitration and verification
- Connection to and dialing of remote instruments using modems
- Socket connections to remote instruments over a LAN
- Industry-standard OSF/Motif graphical user interface.

The first change needed to convert the prototype into a product was to convert the application to use an OSF/Motif™ user interface style. This was done using HP's Interface Architect (also known as UIM/X) user interface management system. The prototype virtual remote had relied entirely on the display library written at the Queensferry Telecommunications Operation for all X display functions. This display library relies on low-level Xlib functions. This approach was adequate for the simulation of the instrument display and front panel. The implementation of application-specific display widgets such as menus and error dialogs was, however, somewhat nonstandard. Interface Architect was used to provide the top-level OSF/Motif application widgets and error dialogs. The low-level display functions using the shared display library were then embedded inside the OSF/Motif application.

The next stage in the development was to split the single-threaded application into constituent parts that could be used to create a multiprocess application capable of supporting the requirements of the final system. The design used is shown in Fig. 7.

Fig. 6. A workstation screen displaying a picture of an instrument drawn by the virtual remote application. All of the data used to draw this image is stored in the instrument and is sent to the virtual remote application when requested. This allows virtual remote to be used with many different instruments without having any special knowledge about how they look or operate.

This design has several advantages. The independent server process is the core of the system. This process is responsible for the arbitration and initiation of all connection requests. If all connections were made using TCP/IP a single server would be responsible for all instruments connected to the network. When using RS-232 for direct or modem connections a server process is required for each workstation. This process is responsible for instruments connected through the RS-232 ports of the workstation upon which it is running. Each server process accepts connection requests from any front-end process. The front-end process may be running on the same host as the server or on any other networked workstation. The separation of the front end from the server gives us the ability to make connections across the network to instruments physically connected to remote workstations. A second advantage of separating the front end from the server is that it allows the provision of more than one front end. At present only two front ends have been provided: an OSF/Motif-based menu interface and a command-line interface. It would be relatively easy to design a new front end, perhaps for integration with some larger system.

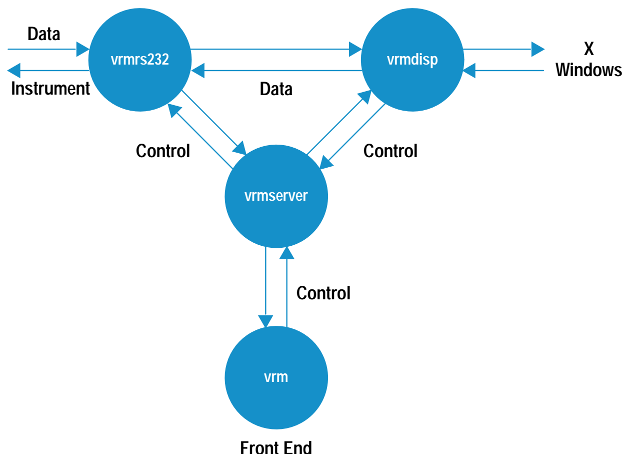


Fig. 7. Virtual remote software architecture.

Once the front end has verified that no one is connected to a particular instrument, it attempts to establish a connection to the instrument. Here again, the communications process has been decoupled from the server. The server must determine the communications channel used for connecting to the instrument. This data is stored in the instrument configuration file. The server then uses this data to start the appropriate communications driver process. The communications drivers provided at present are RS-232 direct connect, RS-232 modem, and TCP/IP. Queensferry Telecommunications Operation instruments have traditionally supported RS-232 and HP-IB (IEEE 488, IEC 625) communications ports. The TCP/IP method of connection is provided to allow networked communications and to allow more simultaneous connections than would be possible using RS-232 ports provided on current workstations. This method operates by using TCP/IP to connect the workstation over the network to a terminal multiplexer. This device maps TCP/IP addresses to RS-232 ports.

The latest generation of instruments produced by the Queensferry Telecommunications Operation can have a ThinLan/Ethertwist interface that allows TCP/IP communication directly to the instrument. This removes the need for the terminal multiplexer and allows any workstation on the network to connect to any instrument.

New communications channels can be provided simply by writing a new driver process.

Once the server has started the communications process and established that the instrument is responding, it starts up the virtual remote display process. This process is the part of the

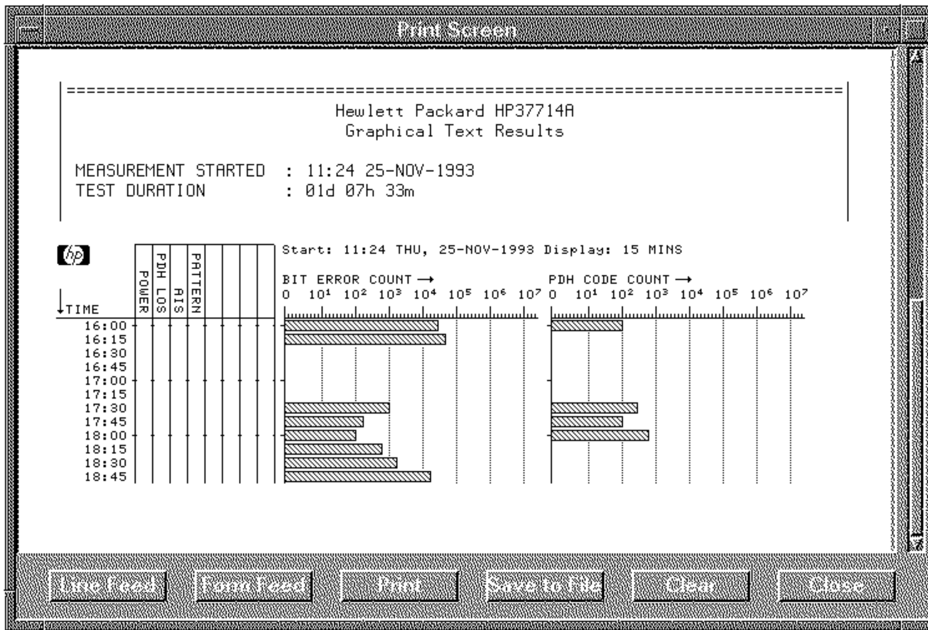


Fig. 8. This is a screen of the virtual printer application. The virtual printer displays print data that the instrument would normally send to a real printer. During virtual remote operation the data is sent to the virtual remote application and is displayed by the virtual printer. The pushbuttons on the virtual printer can be used to send the buffered printer data to a real printer connected to the workstation, save the data to a file, clear the buffer, and so on.

application most closely related to the lab prototype. This process interrogates the instrument for its X display data and uses this data to draw the image of the instrument. It then processes key presses, encodes them, and sends them to the instrument. Data received from the instrument is decoded and the appropriate action taken. This may be the drawing of characters on the screen, illuminating LEDs, or simulating the behavior of an HP ThinkJet printer.

Printer Simulation

The test instruments designed at the Queensferry Telecommunications Operation allow a rich set of data to be logged to an external printer under a variety of operator-defined conditions. An external printer is not of much use if the instrument to which it is connected is not on the same site as the operator using the instrument. It was therefore decided to provide a virtual printer along with the virtual instrument.

Once again this development was facilitated by the ISS development platform. The ISS language/compiler/simulator provides facilities for the instrument designer to compose logging format definitions, which are formatted in the instrument as required. To allow testing of the formats for correctness, an X Windows printer simulator had been developed and integrated with the ISS simulator. The simulator could take the compiled format definitions and simulate their output to an HP ThinkJet printer.

To provide the virtual printer capability to the virtual remote application, all that was required was to integrate the X Windows printer application with the virtual remote application and provide a means to multiplex the printer data with other virtual remote data transmitted between the workstation and the instrument.

The integration of the X Windows printer application with virtual remote was a straightforward task. To maintain the overall look and feel of the application the low-level Xlib code used for the printer application was encapsulated within an OSF/Motif user interface. The separation of the printer data from the standard virtual remote protocol was accomplished by extending the protocol to provide a means of encoding the printer data. This was necessary to prevent raw printer data from clashing with the virtual remote protocol. The instrument logging output driver can detect when the instrument is in the virtual mode of operation and will then output the printer data in encoded rather than raw form. Mutual exclusion is used to prevent the display process and the printer output process from attempting to send data to the workstation at the same time. Using this technique data throughput can be maintained by both processes in a controlled manner.

The virtual printer receives the encoded printer data and decodes it. This data is then displayed in the X Windows HP ThinkJet printer simulation. The simulator can display both text and graphics. At the same time as the printer data is displayed it is appended to a data buffer. This buffered data can be saved to a named file or sent to a printer spooler by the operator. The online buffer can be cleared on demand.

Fig. 8 shows the virtual printer application. The main portion of the application is the "virtual paper," where the text and graphics of the printer data are displayed. A scroll bar along the side of the screen can be used to scroll the paper backwards and forwards. Pushbuttons along the bottom of the screen are used to control the functions described above.

Instrument Customization

In addition to the development of the workstation application code the instrument firmware must also be modified for virtual remote operation. The modifications required affect the following areas:

- Instrument communications drivers (RS-232 interrupt service routine)
- Instrument display library code
- Instrument keyboard processing code
- Instrument external interface selection
- Instrument simulation data.

The first three areas above are contained in the common code distributed to all projects using the Queensferry Telecommunications Operation common platform. The remaining areas are instrument-specific and must be done on an instrument-by-instrument basis.

First, the instrument external interface selection code must be enhanced slightly. A new external connection mode, virtual remote, must be added alongside the existing selections such as printer and remote control. This is a very minor change to the instrument ISS definition file. Two new C functions must be provided. These functions are called by the common code when a virtual connection is being established. These functions are used to make one or two minor configuration changes to the instrument. They cannot be common since they relate to variables defined in individual instrument ISS definition files.

The second instrument customization is the instrument simulation data. In the past the simulation data has only been used internally for instrument simulation during development. In this application it is the simulation of the instrument operator-machine interface that is important, not the quality of the displayed instrument image. The virtual remote application, on the other hand, uses an image of the instrument to make remote operation by a customer easy. In this case the quality of the image displayed is very important. The instrument designer must ensure that the instrument simulation data is as accurate as possible. The areas of concern are instrument dimensions, key positions, text labels, and connector positions and accuracy. Using the syntax of the ISS simulator it is possible to construct a fairly accurate representation of the instrument.

The use of the Queensferry Telecommunications Operation common firmware platform has enabled the virtual remote firmware to be retrofitted in a complete family of instruments with only a small amount of effort. All new instruments developed using the platform will build in virtual remote support from the start.

Summary

Since its release, virtual remote has proved to be a very popular product. A broad spectrum of customers have put virtual remote to work, sometimes in interesting applications:

- A major network operator has installed virtual remote as part of their network monitoring and maintenance system.
- Virtual remote has been used in the installation of cable television networks, allowing centralized testing of the network.
- Virtual remote has been used to test satellite communications channels, allowing simultaneous monitoring of the data links at geographically separate ground stations from a central control center.

The success of virtual remote for HP-UX workstations led to a demand for virtual remote on PCs. HP 15801A PC virtual remote was developed so that as far as possible the source code is shared with the HP 15800A HP-UX product.

Fig. 9 shows an operator using virtual remote on a PC in an office environment. The instrument being controlled is at a remote site and the connection is through a modem and a telephone line.

Fig. 9. The virtual remote software is also available for PCs.

A new generation of instruments designed at the Queensferry Telecommunications Operation have for the first time provided a LAN TCP/IP communications interface. This interface will allow us to have direct network communications between a workstation running virtual remote and the instruments that are being monitored. Almost all advance orders for this new family of instruments have included the instrument virtual remote option. The close links between virtual remote and the common firmware platform allowed the new instrument to be developed with no additional work required to provide virtual remote operation.

The concept of virtual remote has stimulated our customers to find interesting new ways to test their networks. Suggestions for enhancements have come from within the Queensferry Telecommunications Operation and also from our customers. These ideas will be evaluated and may be used to develop virtual remote into a product that can help our customers still further.

Acknowledgments

The author would like to thank everyone who has contributed to the development of virtual remote: past and present members of the Queensferry Telecommunications Operation common firmware group, the engineers responsible for installing virtual remote into individual instruments, and the engineers from our marketing and quality departments. We would also like to thank the engineers of the software house Ascada, who contributed to this program under contract to Hewlett-Packard. Special thanks should go to Malcolm Rix for his work on the first prototype version of virtual remote.

Reference

1. M. Rix, "Case Study of a Successful Firmware Reuse Program," *Proceedings of the 1992 HP Software Engineering Productivity Conference.*

HP-UX is based on and is compatible with Novell's UNIX® operating system. It also complies with X/Open's™ XPG4, POSIX 1003.1, 1003.2, FIPS 151-1, and SVID2 interface specifications.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

X/Open is a trademark of X/Open Company Limited in the UK and other countries.

OSF/Motif is a trademark of the Open Software Foundation in the U.S. and other countries.

Frame Relay Conformance Testing

At HP's Protocol Test Center, an automatic translator was developed to transform abstract test suites into executable test suites for HP IDACOM protocol analyzers.

by Martin Dubuc

The frame relay protocol is a data transfer protocol defined by the American National Standards Institute (ANSI) and the International Telecommunications Union (ITU). It is similar to the ISDN (Integrated Services Digital Network) standard but it assumes a reliable transmission medium and therefore contains very little error recovery functionality. As a result, it is more straightforward and data transfer is more efficient. At present it is used mainly in North America and Japan. A typical application is LAN-to-LAN interconnection.

The frame relay protocol standard leaves many things undefined and so the protocol is sensitive to issues of implementation. To ensure that different manufacturers' implementations are stable and interoperable, standardized test specifications have been developed that thoroughly test the protocol features. The existence of such widely available conformance test specifications greatly benefits the entire frame relay industry. Early availability of frame relay conformance test systems based on these test specifications provides a common reference point for network implementers, ensuring that migration to frame relay technology is as trouble-free as possible.

Hewlett-Packard frame relay conformance test products address the needs of frame relay networks as they exist today and as they will exist in the near future. These products will evolve to comply with the final standardization of the frame relay protocol, thereby ensuring that tomorrow's equipment will be compatible with the networks that are built today.

This article discusses the development of these products. First, we present the ACT-Frame Relay committee. Then we introduce the basic concepts of test suite design and describe the test implementation methodology. We end by presenting the conformance testing environment available on the HP PT502 protocol tester, and the ACT-Frame Relay T1.617 Annex D conformance test product.

ACT-Frame Relay Standardization Committee

In 1991, the Frame Relay Users Forum (FRUF) created a testing and interoperability group. Its main goal was to ensure the interoperability of frame relay devices. The group was further divided into three subgroups responsible for coordinating the development of conformance tests, defining interoperability, and working with test labs to develop baseline tests. The conformance subgroup decided to use the expertise available in the National ISDN Users Forum (NIUF) for the specification of conformance tests. It set up a frame relay group within the NIUF for that matter. This group was named the ACT-Frame Relay committee (ACT-FR).

The ACT-Frame Relay committee's mandate is to develop abstract conformance test specifications for frame relay products in accordance with the relevant standards and the Frame Relay Users Forum implementers agreements.

The primary goal of the ACT-Frame Relay committee is to write the test specification for basic permanent virtual connection (PVC) protocol implementation for customer premise equipment (CPE). It also intends to address network-to-network interface PVC, and, finally, switch virtual connection (SVC).

The ACT-Frame Relay committee also has the mandate to pursue international standardization by presenting its work to the ITU-T, the technical committee of the ITU working on protocol standardization. HP has played an active role so far in this committee and in the specification of the ACT-FR

Glossary

ATS: Abstract test suite, the test specification document.

ACT-Frame Relay (ACT-FR): Frame relay conformance testing standardization committee that was set up by the Frame Relay Forum testing and interoperability group.

CDL: Constraint Description Language, an ITL extension used to specify the coding of the messages exchanged in an abstract/executable test suite.

CPE: Customer premise equipment; a piece of equipment with telecommunication functionality.

ETS: Executable test suite, an implementation of the abstract test suite that runs on a given test platform.

ISDN: Integrated Services Digital Network.

ITL: Interactive Test Language, the built-in language of the HP PT500 protocol analyzer.

ITU: International Telecommunications Union

PVC: Permanent virtual connection.

SVC: Switch virtual connection.

Test case: a test scenario of the abstract test suite. Each test has a narrow test purpose, and test cases with related test purposes are gathered into the same test group.

TTCN: Tree and Tabular Combined Notation, the test specification language standardized within ISO and ITU. TTCN is part of the OSI conformance methodology and framework (ISO 9646).

UNI: User-to-network interface.

T1.617 Annex D UNI (user-to-network interface) and Q.933 Annex A UNI conformance tests. HP currently holds the editorship of these two test specifications.

Abstract Test Suite Design

An abstract test suite (ATS) is a test specification document that describes tests to be carried out to ensure that a particular implementation conforms to a certain specification. The ATS consists of a collection of tests and the definition of the related components that are used for their specification. The basis of a test is its test purpose and the associated dynamic behavior. The test purpose specifies, in an informal language, the behavior of an implementation under test for a given test scenario. The dynamic behavior part of a test specifies the exact ordering of events that must be accomplished to fulfill the test purpose. Events can be classified in four major categories: sending messages, receiving messages, assignments, and timer operations.

An ATS is said to be abstract because it must make an abstraction of the test platform on which the tests will be executed, as well as the higher and lower protocol layers that are used by the protocol layer(s) to be tested.

Most of today's protocol specifications are state-machine driven. A test designer will often use the state table or SDL (Specification and Description Language, CCITT Z.100) diagrams as a basis for defining a set of tests for such protocols. The ATS should test, to the extent practical, the different situations that an implementation might encounter. The ATS should cover all mandatory and optional features of the implementation. It should also test a good sample of the possible inopportune and erroneous events.

Test cases based on a state matrix are specified in a straightforward manner. All the test designer needs to do in such cases is to call a preamble to get the implementation under test to the starting state, then apply a certain behavior to the implementation, and finally, call a postamble, which verifies that the implementation has jumped into the proper state and optionally brings the implementation into a stable, known state.

In TTCN, the test specification language standardized within the ISO and the ITU, some scenarios are more difficult to specify. For instance, it would be nearly impossible to specify in TTCN the exact events that would be needed to test the behavior of an implementation under heavy loads. In this case, the test purpose might be the only thing that a test designer can specify.

Test Suite Implementation

Once an abstract test suite becomes available, it must be converted into an executable form that will run on a specific test platform. For all of its X.25, ISDN, and frame relay conformance testing products, the HP Protocol Test Center uses the T1/WAN protocol analyzers manufactured by the IDACOM Operation of Hewlett-Packard. The strength of these protocol analyzers lies in their flexible programming and conformance testing environments.

All of the executable test suites that were implemented by HP IDACOM before the creation of the Protocol Test Center are written in the Interactive Test Language (ITL). ITL is the built-in language available on HP IDACOM protocol analyzers. It is a Forth-based language with state-machine constructs that allow the implementation of test scripts (sequence of send and receive messages).

ITL is a convenient test script language, but several problems arise with use of this approach to derive executable test suites. First, to obtain executable test suites in ITL, manual coding of the abstract test suite must be done. This process is usually repetitive and error-prone. Another problem is the lack of tools that would speed development by identifying certain types of errors or by ensuring the correctness of the code. Finally, executable test suites written in ITL are very difficult to maintain.

At a certain point, it became obvious that there was a need for automated tools to help implementers develop executable test suites. Because of resource constraints, however, it was decided that in the first phase the tools could not be automated completely.

The most difficult part of an abstract test suite to implement in an executable test suite is the constraint part (message coding/decoding). This is especially true for protocols such as ISDN or frame relay signaling (layer 3) where the content of the messages is fairly complex. A team investigated this problem and implemented an extension to ITL called the Constraint Description Language (CDL). An automatic TTCN-to-CDL translator was also implemented.

This translator was successfully used with some of the ISDN test suites and the early version of the first ACT-Frame Relay test suite. It helped reduce the time required to implement the test suites by a factor of two. However, CDL was not flexible enough to cope with all the constructs used in TTCN. Furthermore, there was room for more automation since a lot of the coding still needed to be done manually.

The next step was to implement a tool that would allow fully automatic translation of an abstract test suite. A team was formed in the Protocol Test Center to investigate how such a translator could be implemented. A tool, referred to as the TTCN translator, was designed to generate executable test suites from abstract test suites written in TTCN. Fig. 1 shows the user interface of the TTCN translator.

Since ITL had limited flexibility, the C language was selected as the target language of the translator. In parallel, a C cross-compiler and a C loader module were implemented. The cross-compiler allows the generation of object files with instructions compatible with the processor of the targeted protocol analyzer (the processor of the HP PT502 is a Motorola 68000). The C loader module allows C object files to be loaded and executed on the HP PT502.

As shown in Fig. 2, to create an executable test suite, a developer first has to provide an abstract test suite. The ATS can be written from scratch using a TTCN editor (such as the one included in the TTCN environment), or the ATS can be one that was written by a standardization body (for instance, the ACT-FR T1.617 Annex D test suite). The TTCN translator takes an ATS and generates a set of files from it. It produces test suite files, a declaration file, a constraint file, a test step file, and test case files.

The test suite files contain information on the different test cases present in the test suite, as well as information on test

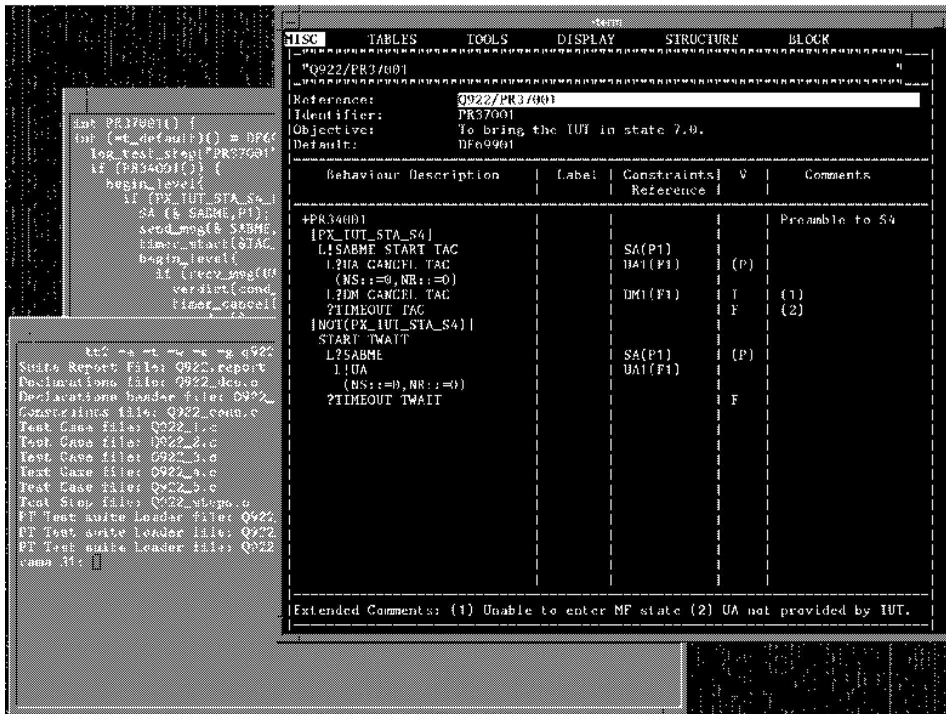


Fig. 1. Conformance testing development environment.

case selection (that is, on which test cases are relevant for testing a specific implementation). The declaration file is a translation of the declaration part of the ATS. It contains all the message structures, timer definitions, and test suite constants, variables, and parameters. The constraint file is the translation of the constraint part of the ATS. It contains the exact coding of all messages to be sent or received. The test step and test case files are the translation of the dynamic behavior of the ATS. The dynamic behavior specifies the ordering of send, receive, assignment, and timer events to

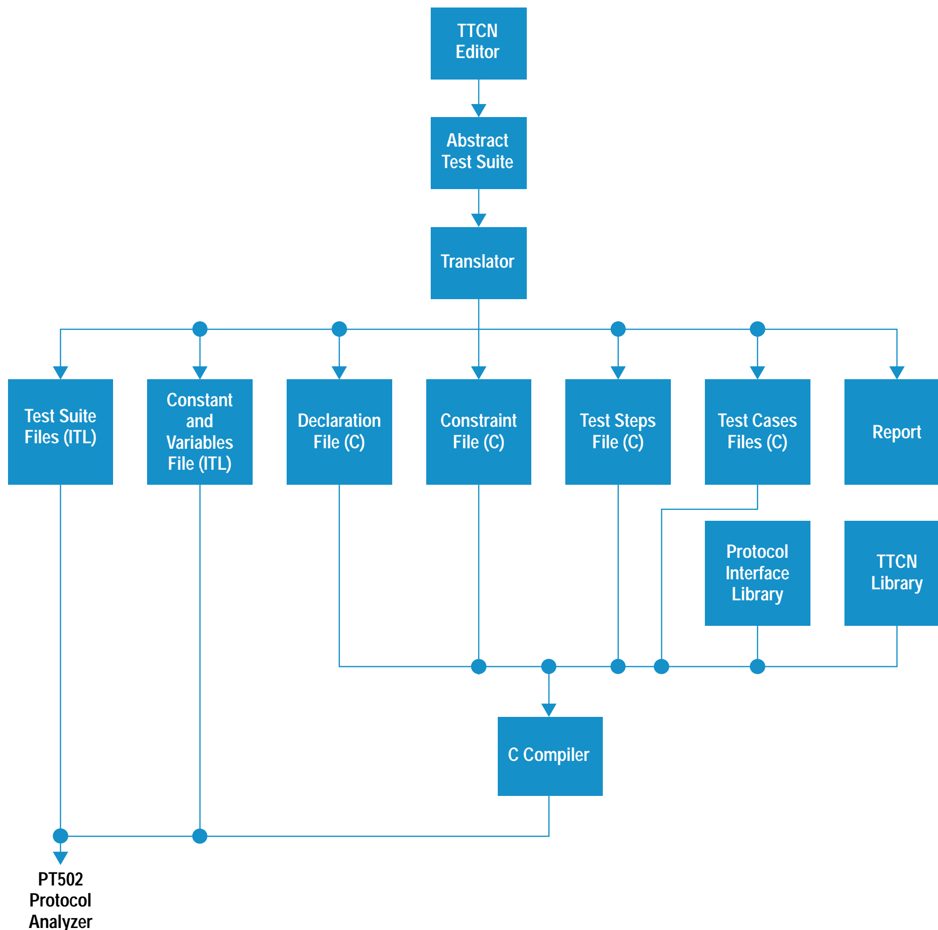


Fig. 2. Executable test suite product life cycle.

```
AP #1:    Frame Relay Emul        Playback RAM       1993–11–05 13:11:53
 Source  DLCI   C/R FECN BECN DE  Info Field
Test Case: PS0_01V STATUS ENQUIRY     Request   1993–11–05 13:10:02
H1 RX   0      0      0     0      0 STATUS_EN
H1 TX   0      0      0     0      0 STATUS
H1 RX   0      0      0     0      0 STATUS_EN
H1 TX   0      0      0     0      0 STATUS
H1 RX   0      0      0     0      0 STATUS_EN
H1 TX   0      0      0     0      0 STATUS
H1 RX   0      0      0     0      0 STATUS_EN
H1 TX   0      0      0     0      0 STATUS
H1 RX   0      0      0     0      0 STATUS_EN
H1 TX   0      0      0     0      0 STATUS
H1 RX   0      0      0     0      0 STATUS_EN
Test Case: PS0_01V STATUS ENQUIRY     Request      Verdict: PASS(0)
>
 ----------------------------------------------------------------------
 rch  ResponseTime  Print  Filters  Triggers  TestScript  TestKeys  TestSuite
 ----------------------------------------------------------------------
|      |f1|      |      |f2|        | |f3|  |  |f4|  |     |f5|     |    |f6|     |    |f7|   |
| Load Suite  | Tester Setup  | PICS | PIXIT  | Selection  | Run Suite  | Reports  |
```

Fig. 3. Conformance testing environment.

specify formally the test purpose of every test case of the ATS.

The TTCN environment provides protocol interface libraries and TTCN libraries, which are necessary to interface the translated code with the protocol tester used to run the executable test suites (HP IDACOM protocol testers in our case). The protocol interface libraries specify, for a specific protocol, what functions of the protocol analyzer are used to send and receive a message and how to initialize the lower layers at the beginning of a test campaign. These parameters are specific to the protocol to be tested, whether the test suite is a frame relay or an ISDN test suite.

Once the files are generated, the C files are compiled using the cross-compiler provided in the TTCN environment, and the resulting object files are linked together along with the TTCN and protocol interface libraries. The test suite files and the object file are then sent to the HP PT500 and the executable test suite (ETS) is ready to be loaded into the conformance testing application.

The TTCN translator has been used internally for a year now, and the results have exceeded expectations. The time needed to implement test suites has been reduced by nearly an order of magnitude or close to ten-to-one. Furthermore, it is now possible to address more complex test suites, such as test suites meant to test a network.

PT502 Conformance Testing Environment

The screen display of the HP PT502 running the ACT-FR T1.617 Annex D ETS is shown in Fig. 3. The conformance testing environment is the environment in which the user, also called the test operator, sets up and launches a test campaign and generates test reports. Several services are available to the user in this environment.

First, the test operator can specify the different system parameters that characterize the implementation under test using the PICS and PIXIT screen menus. This will help the protocol tester identify which test cases are relevant to a specific implementation.

Once the configuration is done, the test operator can set up a test campaign by selecting the test cases to be executed. Test cases can be selected through the test group and case selection menus using different criteria: select a whole test suite, select/deselect a whole test group, select/deselect individual test cases, select by verdict assignment (for instance, select all test cases that previously failed).

During the execution of a test campaign, the protocol tester records verdicts for each test that it runs. In addition, traces can be recorded for each test, or they can be recorded according to the verdict assigned to the test. For instance, the traces of all failed tests can be recorded.

When a test campaign is finished, a test report can be printed. The tester reports information about the implementation configuration, as well as a summary of the campaign, which includes the verdicts for each test. It can also report the full traces of tests with given verdicts.

Traces can be displayed, printed, or played back using different formats. The format menu can be used to specify whether the traces should be displayed in hexadecimal, short, or complete forms. Optionally, if the complete format is specified, the decoding of each individual information element can be done. The time stamp can also be added in the trace format. In Fig. 3, the short format was used. Fig. 4 shows the trace of a STATUS message with complete decoding and full message detail.

ACT-FR T1.617 Annex D Test Suite

The ACT-FR T1.617 Annex D ATS was the first ATS specified by the ACT-Frame Relay committee. This test suite was aimed at testing the permanent virtual connection (PVC) management procedures for the user-to-network interface (UNI). It covers the following areas:

- Notification of the addition of a PVC
- Detection of the deletion of a PVC
- Notification of availability or unavailability of a configured PVC
- Link integrity verification.

Table I shows the test suite structure for the ACT-FR T1.617 Annex D ATS. The two main groups (periodic polling and bidirectional) are each further divided into three subgroups (general, error, and system).

Table I
ACT-FR T1.617 Annex D Test Suite Structure

Test Group Name	Number of Test Cases in Test Group
Periodic Polling/General	24
Periodic Polling/Error	42
Periodic Polling/System	3
Bidirectional/General	9
Bidirectional/Error	13
Bidirectional/System	1

The periodic polling group contains the test cases related to the testing of the user-side link integrity verification procedure, while the bidirectional group contains test cases related to the testing of the bidirectional network procedures. The latter procedures are usually implemented on network equipment (and thus can be used to test a network-to-user interface), but can be supported optionally by a user's equipment.

```
MM:SS:ssss Source  DLCI  C/R   FECN  BECN  DE  Info    Field
18:40.2420   H1 TX    0     0     0     0     0   UI      P=0

    PD = T1.617  Dummy Call Reference        STATUS
 1   1001----  INFORMATION ELEMENT       :  SHIFT
     ----0---  Shift type                :  locking
     -----101  Codeset ident.            :  national use IE
 1   00000001  INFORMATION ELEMENT       :  REPORT TYPE
 2   00000001  IE length                 :  1 octet
 3   00000000  Report Type               :  full status message
 1   00000011  INFORMATION ELEMENT       :  LINK INTegrity VERification
 2   00000010  IE length                 :  2 octets
 3   00001000  Send Seq Number           :  8
 4   00010111  Rcvd Seq Number           :  23
 1   00000111  INFORMATION ELEMENT       :  PVC STATUS
 2   00000011  IE length                 :  3 octets
 3   0-------  Extension bit             :  continued
     -0------  Spare                     :  don't care
     --000001  PVC DLCI                  :  partial value
3A   1-------  Extension bit             :  not continued
     -0000---  PVC DLCI                  :  16
     -----000  Spare                     :  don't care
 4   1-------  Extension bit             :  not continued
     -000----  Spare                     :  don't care
     ----0---  PVC Status                :  PVC is already present
     -----0--  Spare                     :  don't care
     ------1-  PVC Active                :  PVC is active
     -------0  Spare                     :  don't care
 1   00000111  INFORMATION ELEMENT       :  PVC STATUS
 2   00000011  IE length                 :  3 octets
 3   0-------  Extension bit             :  continued
     -0------  Spare                     :  don't care
     --000001  PVC DLCI                  :  partial value
3A   1-------  Extension bit             :  not continued
     -0001---  PVC DLCI                  :  17
     -----000  Spare                     :  don't care
 4   1-------  Extension bit             :  not continued
     -000----  Spare                     :  don't care
     ----0---  PVC Status                :  PVC is already present
     -----0--  Spare                     :  don't care
     ------0-  PVC Active                :  PVC is inactive
     -------0  Spare                     :  don't care
 1   00000111  INFORMATION ELEMENT       :  PVC STATUS
 2   00000011  IE length                 :  3 octets
 3   0-------  Extension bit             :  continued
     -0------  Spare                     :  don't care
     --000001  PVC DLCI                  :  partial value
3A   1-------  Extension bit             :  not continued
     -0010---  PVC DLCI                  :  18
     -----000  Spare                     :  don't care
 4   1-------  Extension bit             :  not continued
     -000----  Spare                     :  don't care
     ----1---  PVC Status                :  PVC is new
     -----0--  Spare                     :  don't care
     ------1-  PVC Active                :  PVC is active
     -------0  Spare                     :  don't care
18:40.2422
```

Fig. 4. Trace display of a STATUS message using the complete format.

The general subgroup gathers the test cases related to valid events. In these test cases, the tester simulates a normal exchange of messages. The error subgroup contains test cases that simulate network equipment that does not conform to the standard and verifies that the implementation reacts according to the specification. In these test cases, the tester sends messages with content error, and verifies that the implementation under test rejects either the whole message or the erroneous part. The system subgroup contains system tests and tests that verify the system parameters of the implementation under test.

It is worth noting that often part of a standard is too vague. How should the equipment react under such circumstances? An ATS can sometimes provide additional guidelines on these points, and the ACT-FR T1.617 Annex D ATS is a good example. The implementation behavior under certain error conditions is not specified in the standard. The ACT-FR committee has worked jointly with the T1S1.2 working group to fill in the holes in the standard. The ATS was specified according to the results of this process. The T1S1.2 group plans to update the standard accordingly, but in the meantime only the ACT-FR ATS provides the intended interpretation of the standard.

The ACT-FR T1.617 Annex D ETS for the HP PT502 can be used for different purposes. It is an excellent tool for frame relay equipment implementers for pretesting and preacceptance purposes. The implementer can use the ETS at different stages of development—for instance, to test features incrementally as they are implemented. The ETS can also be used for regression testing, to ensure that changes or the addition of new features to the frame relay equipment have not introduced deficiencies in the implementation. The ETS tests most aspects of the specification.

The test suite can also be used by carriers to test frame relay equipment before it is hooked to a network (acceptance testing). In this way, the carriers can ensure a certain degree of interoperability among the different instruments of their network.

Finally, along with other analysis packages available on the HP PT502, the ETS can be used for troubleshooting by network managers.

Future Directions

ACT-FR is now ready to tackle its next assignments, namely, development of additional procedures for permanent virtual connection (network-to-network interface), switch virtual connection data link layer (LAPF), and switch virtual connection signaling. Hewlett-Packard intends to continue its role in the ACT-FR committee and to release an ETS for each ATS approved therein.

Conclusions

Use of conformance testing in the early steps of the product development life cycle speeds overall time-to-market of communicating devices. It also increases the probability of interoperability between products from different vendors.

In the past, conformance test software was coded manually. The process was repetitive and error-prone. In today's competitive market, these methods are not good enough to provide customers in a timely fashion with conformance test software to be used in-house for pretesting.

The HP Protocol Test Center R&D team has implemented a set of productivity tools to automate the implementation of executable test suites. These tools have dramatically reduced the R&D efforts needed to produce executable test suites. They have also enhanced the quality of the software and contributed to the detection of problems in the test specifications by underlining inconsistencies.

The ACT-Frame Relay committee has adopted the test specifications for the additional procedures for permanent virtual connections (user-to-network interface). With the help of the TTCN translator, Hewlett-Packard was the first company to offer a product for those test specifications.

The FDDI Ring Manager for the HP Network Advisor Protocol Analyzer

The FDDI Ring Manager application takes the knowledge burden from the user and puts it on the network management tool. It pulls ring status information from station management frames and presents it in a logically ordered display. It gathers ring topology information from neighbor information frames and status information frames and presents that information in a graphical map and a textual report.

by Sunil Bhat, Robert H. Kroboth, and Anne L. Driesbach

FDDI (Fiber Distributed Data Interface) networks offer high-speed data transfer and fault-tolerant dual-ring topology, but can add a layer of complexity to troubleshooting network problems. The dual ring can become twisted or wrapped, a serious problem that may not be immediately apparent. Interoperability issues can cause ring abnormalities between products from different vendors, but tracing these glitches to their source by looking at decodes can be intensely time-consuming. And, although the FDDI station management protocol defines a management information base (MIB) full of interesting information, it may not be convenient to access and make sense of that information. The FDDI Ring Manager application for the HP 4980 Series Network Advisor protocol analyzers[1] addresses these and other FDDI network management issues in one integrated network management application.

Fig. 1 shows the FDDI Ring Manager user interface. The current state of the dual ring and general network performance are reported along with other interesting network information in the top area of the window. The middle area of the window is a graphical map of the FDDI network. The nodes on the map can be opened to examine station management MIB information. The FDDI Commentator, which is automatically started when the FDDI Ring Manager application is started, reports trends, changes, and events of interest.

The design of the FDDI Ring Manager is based on customer input. Early in the investigation stages of the FDDI Ring Manager, we decided to use five customer sponsors as development partners. After the first several iterations with the five sponsors, customer requirements could be summarized as:

- Make it easy to see what's happening on the network

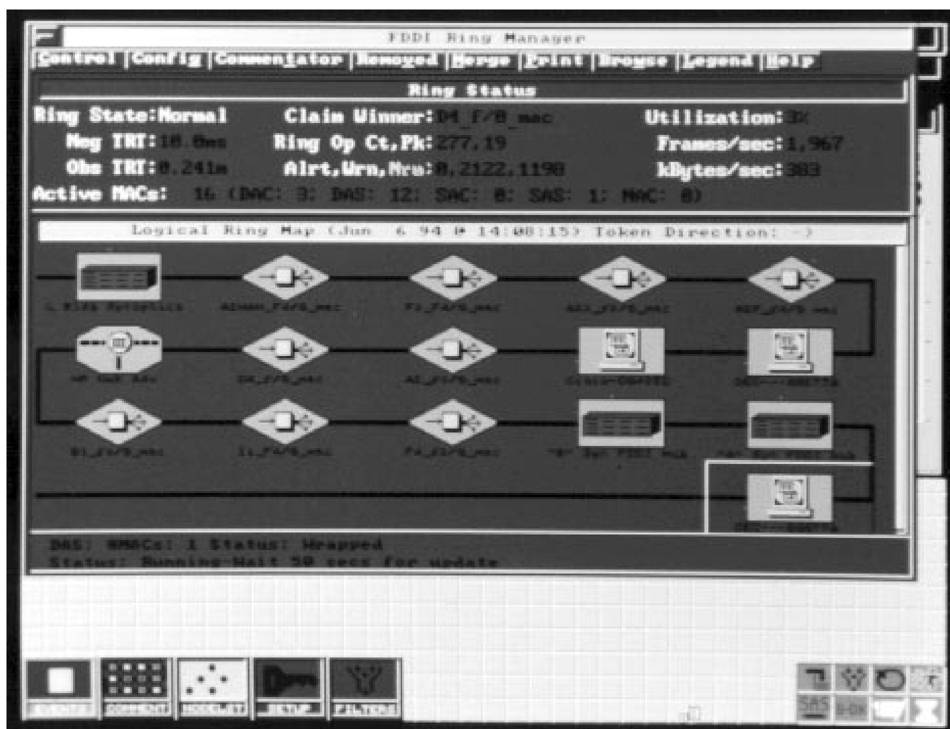


Fig. 1. FDDI Ring Manager user interface.

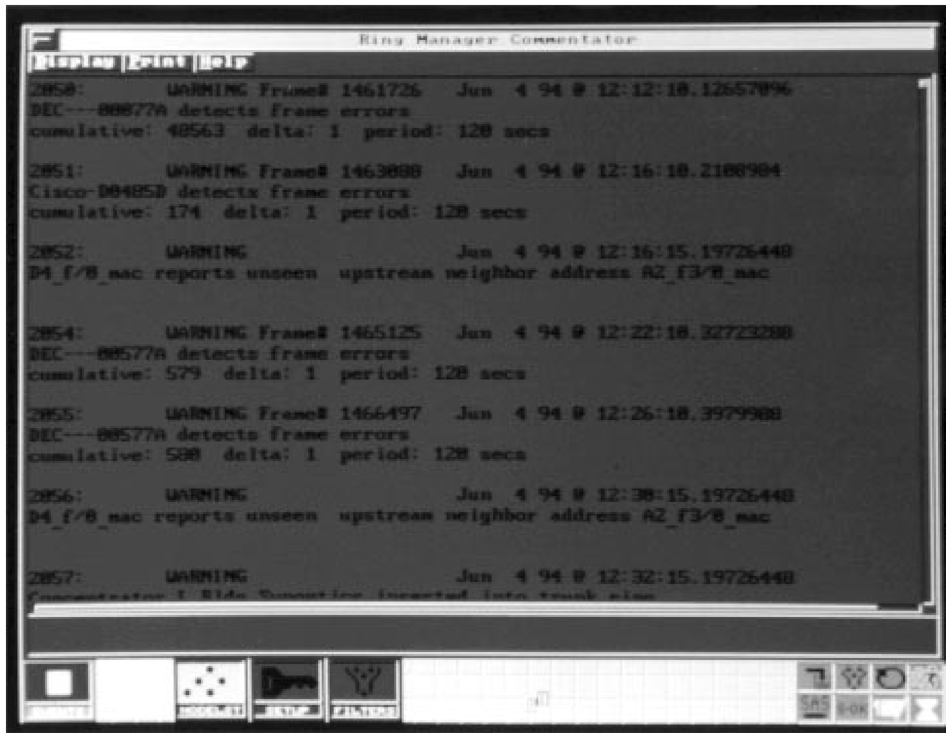


Fig. 2. FDDI Commentator window.

- Make it easy to see where network problems are
- Automate tedious tasks
- Provide a tool that does not require the user to be an expert on the Network Advisor and an expert on FDDI protocols.

As a result of ongoing customer consultations, we answered these requirements with the following FDDI Ring Manager features:

- A ring status section to give a high-level overview of the ring state
- Physical and logical ring maps to show graphical network topology
- Dynamic error indication using colors to isolate problem nodes
- Automated polling of each station management MIB for proactive indications of marginal conditions
- Station information windows to give an orderly presentation of station management MIB information
- The FDDI Commentator to provide a history of events, trends, and changes on the FDDI ring
- Ease of building a node list provided by a link from the FDDI topology maps to the node list.

The key contribution of the FDDI Ring Manager is to take the knowledge burden from the user and put it where it belongs, on the network management tool. Instead of requiring the user to sift through station management decodes looking for fields that may relate to the status of the ring, the FDDI Ring Manager pulls that information from station management frames and presents it in a logically ordered display. Rather than expect the network manager or operator to look at neighbor information frames and figure out which stations are upstream and downstream of a particular station, the FDDI Ring Manager presents that information in a graphical map and in a textual report. Why require the user to keep track of changes in link error rates for particular stations or trends of claiming† by one or more stations? This information can be collected and synthesized into meaningful events that are logged in the FDDI Commentator. Fig. 2 shows the FDDI Commentator window.

FDDI Ring Manager Overview

The FDDI Ring Manager window is divided into three panes: the ring status section, the ring maps or topology section, and the status bar.

Ring Status. The top section of the FDDI Ring Manager window is the ring status section. It provides high-level network health information. Some of the indicators in this section are:

- Ring State: Shows whether the ring is in a normal, twisted, or wrapped state.
- Observed Token Rotation Time (TRT): This is the actual time interval between successive tokens.
- Active MACs: This is a list of currently active media access control units (MACs) for each FDDI station type.
- Network Traffic: Shows the current level of network traffic displayed as a percentage of total bandwidth, and in frames per second and kilobytes per second.
- Ring Op Statistics: This is the number and peak rate of ring initializations in any five-second interval since the start of the FDDI Ring Manager measurement.
- Commentator Events: This field shows how many events of each level of severity have occurred and how many were logged to the Commentator.

Ring Maps. The next section of the FDDI Ring Manager window is the ring maps. The direct-manipulation graphical network map shows the token path through the ring in the logical ring map or the hierarchy of the physical connections

† Claiming is a bidding process to determine the token holding time for ring stations. The claim process is initiated when a station enters the ring (inserts) or leaves the ring (deinserts), but it can also be the result of media or configuration problems.

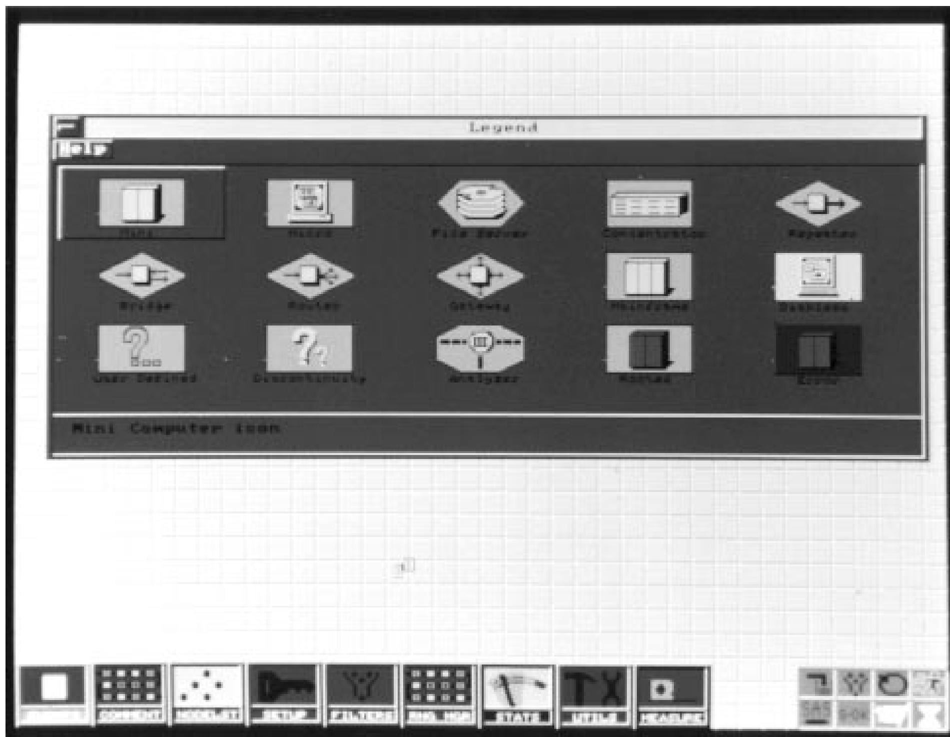


Fig. 3. Legend window.

in the physical ring map. Nodes on the graphical maps turn red or yellow to give dynamic error indication of problems. You can open or "drill down" into the problem nodes or any selected node on the map to see the current configuration and station management MIB values.

Different node icons represent the different device types on the ring and colors are used to communicate the status of each device (see Fig. 3):

- Blue: Trunk (dual-ring backbone) station
- Cyan: Node connected to the ring through a concentrator
- Yellow Concentrator: Indicates a problem in the subtree of the concentrator
- Yellow Station: Indicates that this node has detected frame errors on the ring
- Red: Indicates an alert-level problem with this station or concentrator.

Status Bar. The status bar is a two-line area at the bottom of the FDDI Ring Manager window. It shows the high-level status for the highlighted node. For example, the status may read "Node is wrapped on Port B."

Station Information

Information for every station on the network can be examined by selecting that station's icon on the ring map. Even stations that have left the ring since the FDDI Ring Manager was started can be examined in the removed MACs window (Fig. 4). The state of a removed station's MIB can provide valuable clues when troubleshooting a segmented ring.

When the physical ring map is displayed, concentrators can be opened to show the M-port† connections beneath them. Any concentrator connected to the M-port of another concentrator can be opened to its own M-port map, thus showing the hierarchy of physical connections (Fig. 5). Station information windows for concentrators are accessed with a menu selection (Fig. 6).

The station information display for both stations and concentrators is composed of the following sections:

- Station Description. This section reports the station type, the number of MACs in the station, the number of M-ports, and the upstream and downstream neighbor addresses of the station.
- Port Status. This section shows the state of the connection between each local port and its remote port, the link error monitor reject count for each port, the link error estimate for each port, and the time the port connected to or disconnected from the ring.
- Frame Statistics. This section shows frames transmitted, received, not copied, or lost for each MAC in this station.
- Timer Values. This section shows the station's requested token rotation time, the negotiated token rotation time, the valid transmission timer setting, the maximum token rotation time, and any allocated synchronous bandwidth.
- Miscellaneous. This section shows any user or manufacturer data for this station.

The FDDI Commentator

The FDDI Commentator runs automatically when the FDDI Ring Manager is started. It provides a real-time commentary on all significant events occurring on the ring. These include events that lead to changes in ring topology, events that indicate a problem on the ring (or conditions that might lead to a problem), and other normal events that are likely to be of interest to a network administrator. The FDDI Commentator performs the task of combing through decoded frames for information of interest, a task usually left to the user.

The events in the FDDI Commentator window are collected over multiple runs of the FDDI Ring Manager. The contents

† Network nodes can have four types of ports: A, B, S, and M. M-ports form one end of tree mode connections, that is, they are used to expand the network tree. See article, page 97.

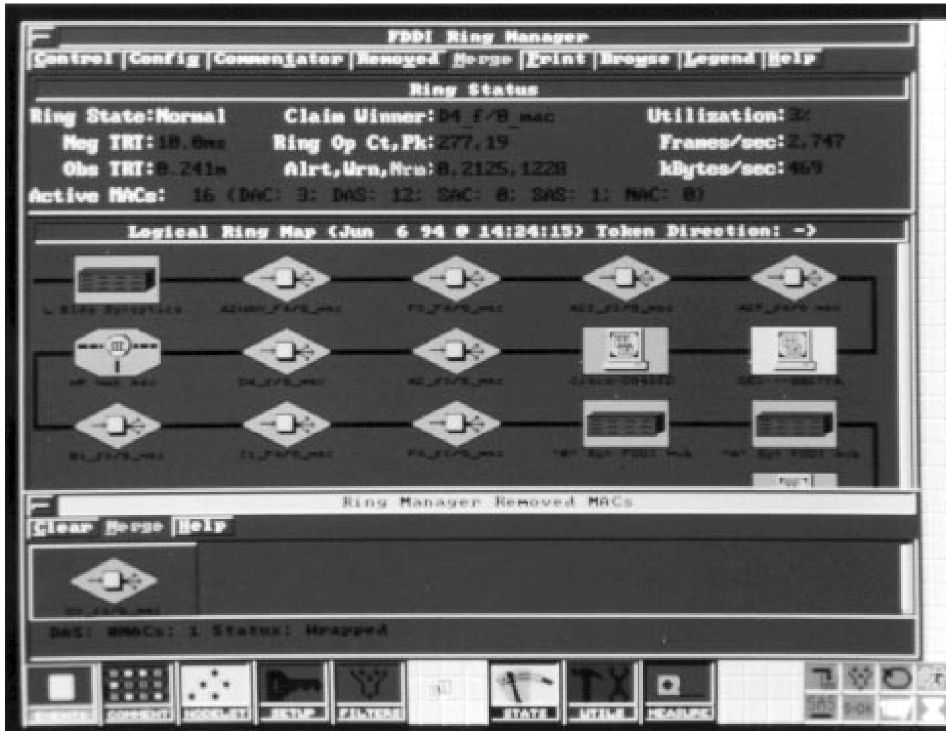


Fig. 4. Removed MACs window.

of the FDDI Commentator can be printed to a file or to the printer.

High-Level Architecture

A quick overview of the architecture of the FDDI version of the HP Network Advisor protocol analyzer will provide a context for a discussion of the architecture of the FDDI Ring Manager. The software architecture of the FDDI Network Advisor consists of three major subsystems. Each subsystem operates on a separate hardware unit with its own processor as shown in Fig. 7. Each subsystem constitutes a multiprocess environment that supports the execution of multiple software modules. The environments are (1) general-purpose, (2) analysis and real-time, and (3) station management.

The general-purpose environment is responsible for a Smalltalk-based windows graphical user interface and related user interface activities. It maintains and manages persistent information such as node lists and setup configuration. The general-purpose environment communicates with the analysis

Fig. 5. M-port window.

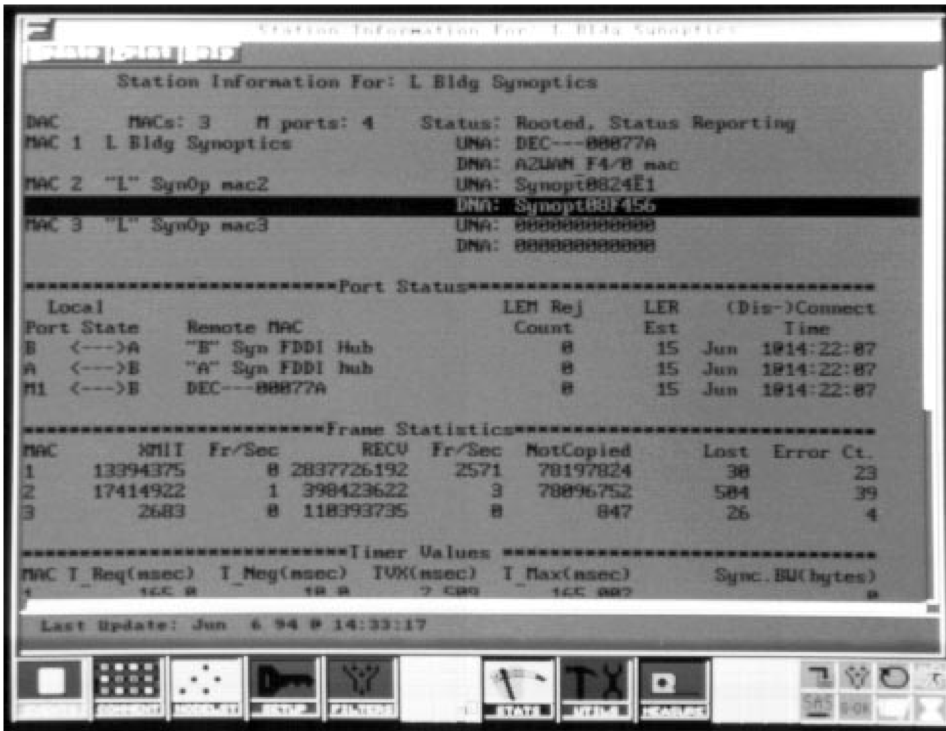


Fig. 6. Station information window.

and real-time and station management environments using commands sent via the interenvironment process communication (IEPC) channel.

The analysis and real-time environment is an event-based nonpreemptive system optimized for network protocol analysis and troubleshooting. It responds to commands from the general-purpose environment and sends back the results of its analysis in specially packaged units called analysis data units (ADUs).

The station management environment is unique to the FDDI version of the Network Advisor protocol analyzer. Its primary function is to run the station management (SMT) process required by the FDDI standard for all stations on the ring.

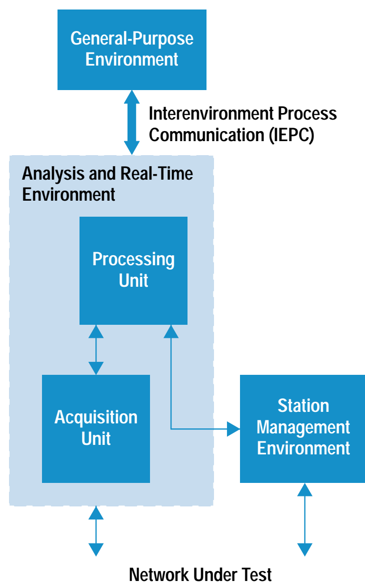


Fig. 7. FDDI Network Advisor architecture.

The implementation of the station management process was acquired from Distributed Systems, Inc. The analysis and real-time environment and the station management environment communicate via the station management interface. The FDDI Ring Manager software spans all of the above environments. It consists of software modules in the general-purpose environment that implement the user interface, and in the analysis and real-time environment that provide the basic analysis engine. The analysis engine uses ring information from the station management environment as maintained by the station management process. The data flow diagram for the entire application is shown in Fig. 8. A discussion of the FDDI Ring Manager software modules in the general-purpose environment and the analysis and real-time environment follows.

General-Purpose Environment GUI Subsystem

The FDDI Ring Manager graphical user interface (GUI) subsystem is an object-oriented system written in Smalltalk. This system heavily leverages the existing HP Network Advisor software platform to provide the user interface functionality for the FDDI Ring Manager. The FDDI Ring Manager user interface subsystem is designed to process various FDDI Ring Manager analysis data units (ADUs) sent from the analysis and real-time environment and display these events appropriately. In addition, control and user commands are processed by the user interface subsystem, and if needed, they are sent to the analysis and real-time system. Fig. 8 shows the data flow diagram for the user interface subsystem.

FDDI Ring Manager View. The FDDI Ring Manager view controls the display of three types of data in the top-level FDDI Ring Manager window (see Fig. 1). This view contains three subviews—one for each section, or tile, of the top window. As the view receives data, it checks its type and routes the data to the appropriate subview. When a subview receives

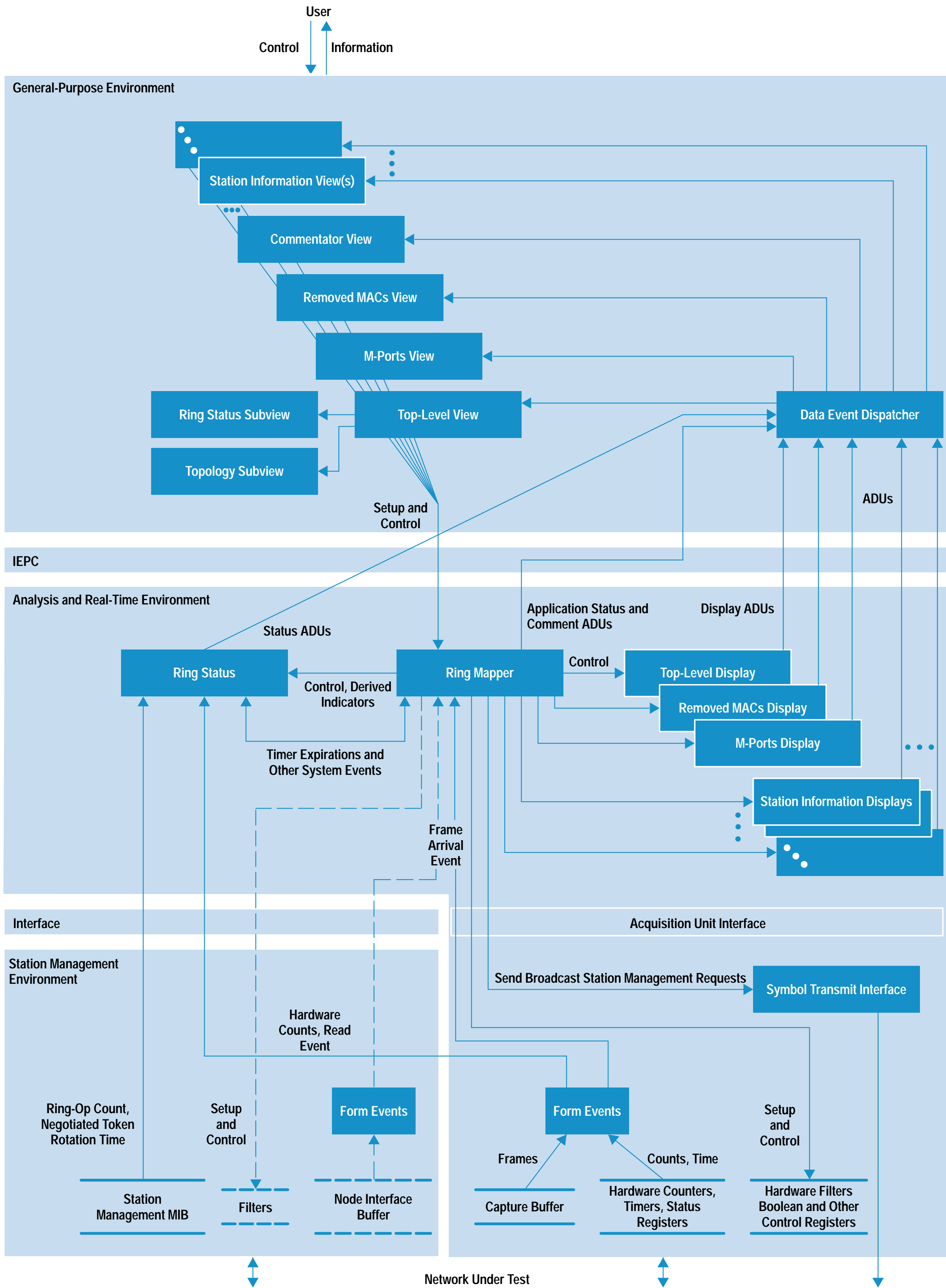


Fig. 8. FDDI Ring Manager data flow diagram, including graphical user interface subsystem data flow diagram. The dashed lines indicate the node interface data path.

the data object, it is responsible for processing the data and displaying it in its tile.

The top and bottom subviews in the FDDI Ring Manager view process ring status and analysis and real-time status messages and display the data in these messages in their tiles. These tiles are read-only and require a simple translation from analysis and real-time data to ASCII data before the data is displayed. The subview that controls the middle tile, the topology subview, is discussed below.

Ring Topology Subview. The ring topology subview is responsible for the mapping of the ring topology data objects (representing MACs on the network) in the middle tile of the FDDI Ring Manager window. This subview also provides generic topology display functionality that is subclassed and used by the M-port connections topology subviews (see Fig. 5) and the removed MACs topology subview (see Fig. 4). Each topology subview must work closely with its corresponding display module in the analysis and real-time environment to provide user interface navigation and "drill-down" capability for the user (see below).

Topology Mapping. When a data ADU is sent to the topology subview, it contains the information for each MAC that must be displayed in the tile. For each MAC, the subview must determine which icon needs to be displayed based on the type of network device the MAC represents (Fig. 3 shows all of the icons used by the FDDI Ring Manager). This is done by first looking up the MAC in the node list using the MAC's address (found in the ADU). If there is a match, the node list type is used. If there is no match, the generic type (concentrator or station) found in the data ADU for the MAC is used. A lookup is then done in the FDDI Ring Manager's icon dictionary to retrieve the appropriate icon for display based on the MAC's type.

The graphical icons used in the display are prebuilt and are accessed through an FDDI Ring Manager icon dictionary. The icon dictionary contains associations of the form (type, icon) where type is the key and icon is the value. The icon dictionary is stored on disk and loaded when the FDDI Ring Manager measurement is instantiated.

Once the icon is obtained, the connecting lines are drawn to the icon based on connection information included in the ADU. If a MAC has an error or warning status, or if the MAC is rooted (on the dual ring), the icon is masked with the appropriate color (red, yellow, dark blue). After all MACs have been processed, they are painted to the screen.

User Interface Navigation. A topology subview controls a tile that displays icons for up to 25 MACs at any one time. If the ring has over 25 MACs (an FDDI ring can have up to 500 nodes), a different set of MACs can be displayed. This minimizes the amount of data that needs to pass through the IEPC for a topology update. The scroll bar or cursor control keys can be used to display different sections of the ring.

When the user sizes the window, thus changing the topology tile size, or navigates to inspect other MACs on the network, commands are generated for the subview's corresponding display module and sent from the subview to the view. Examples of commands are: previous page, next page, previous line, next line, home, and end. When the view receives a command, it adds its handle and sends it to the measurement. The measurement packages it and sends it to the IEPC, destined for the subview's corresponding display module. If the display module determines that the user interface should be updated to reflect the command, it generates a display ADU for the view and sends it to the IEPC. The display ADU is routed to and displayed by the topology subview that sent the command.

"Drill-Down" Capabilities. The act of selecting and opening an object to get detailed information is termed "drilling down." Each MAC icon in a topology subview is a control object that can receive and process user input. The user can move the cursor over the icons and select one by either clicking with the mouse or pressing the enter key while the MAC icon is highlighted. When a MAC icon is selected, the user interface determines if an existing window should be made the top window or a new window should be opened. In either case, the window that is displayed as a result of the icon selection displays additional information about the MAC for the user.

If the user interface determines that a new window needs to be opened it creates the view, registers it with the data event dispatcher, and sends a command to the analysis and real-time environment to create a display module of a particular type. The type of display module created depends on the type of view needed. If the selected node is a rooted concentrator and the FDDI Ring Manager is configured for physical mode operation, an M-port connections view is created and a "create M-port connections display module" command is sent to the analysis and real-time environment. In all other cases, a station information view and a station information display module are created. After the create command is sent to the analysis and real-time environment, the newly created display module sends a display ADU back to the view's subview and the data is displayed. Figs. 5 and 6 show the windows controlled by these views.

The Ring Commentator View. The ring commentator view controls the window that displays the commentator events sent to it by its corresponding analysis and real-time display module. The commentator view contains a subview that is responsible for the processing and display of each commentator ADU sent from the analysis and real-time environment (See Fig. 8).

Each ADU sent to a commentator view consists of a comment type, one to four line identifiers, and zero to four arguments for each line. Arguments are values used when the ADU is displayed. Examples of arguments are MAC address, port identifier, and error count. When the subview receives an event from the view, it looks up the correct ASCII string and event level for the event based on its comment type. This string contains a one-line description for the event, suitable for posting to the Network Advisor's event log. The event level determines the event severity. The severity can be either normal, warning, or alert. The file containing the event levels is ASCII and the levels can be edited by the user. Any changes to the event levels are reflected in the FDDI Ring Manager the next time it is executed.

Each line of text for the event is then found by looking it up in a file based on the line ID. The line found is an ASCII string resembling the C-language printf format string. Each line is converted to the proper display string using the format

characters to control the translation of each of the line's arguments. When each line is translated, it is displayed in the subview's tile.

Analysis and Real-Time Subsystem

Ring Status Module. The ring status module is responsible for monitoring the key indicators of network health. It is controlled by the ring mapper module which is also the central point of control for the FDDI Ring Manager application as a whole. As described earlier, these indicators come from various sources within the analysis and real-time and station management environments.[2] From an implementation point of view the indicators can be categorized as basic or derived. Basic indicators are those that are either measured and maintained by hardware, such as observed token rotation time or network utilization, or available as part of the normal FDDI station management process, such as negotiated token rotation time and ring op count. Derived indicators are those that are outputs of other software modules. They tend to be inferred or derived from relevant station management frames (neighbor information frames and/or station information frame responses). Examples of this type of indicator are ring state, active MAC statistics, claim winner, and metastatistics on the commentator events.

The data flow diagram of the FDDI ring manager in Fig. 8 shows the ring status module in relation to the rest of the FDDI Ring Manager software in the analysis and real-time environment. The operation of the ring status module is driven by timers for the most part. At regular intervals (five seconds for basic indicators and longer for others), the ring status module obtains the values for these indicators, packages them in an ADU and ships it to the general-purpose environment.

Since the status consists of numerous indicators, some maintained in hardware and others in software, we faced the issue of having to combine indicators that reflect two different points in time. This is because the software takes time to process data and therefore trails the hardware by a few milliseconds. We solve this by reading the hardware-maintained indicators along with the real-time hardware clock at regular intervals. When the software finishes processing data corresponding to real time T, the hardware indicators corresponding to time T are combined with the software indicators for time T and the combined indicators are posted as the status for time T. In this way, time consistency is ensured for all ring status pane statistics.

Ring Mapper Module. The ring mapper module forms the core engine for the FDDI Ring Manager software in the analysis and real-time environment. It implements the algorithms for building logical and physical views of the ring. These algorithms are described in detail in the article on page 97. The ring mapper also maintains the list of removed MACs and generates event notifications for the FDDI commentator.

The ring mapper uses the front-end monitor data path to obtain neighbor information frames and station information frame responses to its broadcast configuration and operation station information frame requests. Optionally, we could have chosen to use the node interface data path in the station management environment. This is shown in Fig. 8 using dashed lines.

Using the station management environment data path would have required Distributed Systems, Inc. to make changes to its station management software to support our requirements. Since this would add a dependency beyond our control for the project, we decided against this option. This decision was not without its downside. For performance reasons the ring mapper sets a station management filter to allow only station management frames to be saved in the capture buffer that feeds the monitor data path. This filter precludes other applications that use nonstation-management frames for their analysis from running simultaneously with the FDDI Ring Manager. However, we left open the possibility for change by implementing the ring mapper so that the source of its information is completely irrelevant to its operation.

The ring mapper compiles and organizes all necessary information from the station management frames it receives. From this information, it determines the ring maps and provides on demand detailed station information for all nodes on the ring. For the physical view of the ring, the ring mapper determines physical links between stations on the ring and to which ports that they connect.

All general-purpose commands to control the entire FDDI Ring Manager application in the analysis and real-time environment are sent to the ring mapper, which may act on a command or redirect it to the appropriate module—for example, a display module.

Display Modules

A display module is responsible for supporting a virtual window in the general-purpose environment. There is a display module for each open window in the FDDI Ring Manager application. A window's display module supports standard windowing operations such as next page, previous page, next half page (for scrolling down), previous half page (for scrolling up), show page starting at specified point in virtual space (for home, end, etc.), and update page.

All display modules get their information from the ring mapper. They send their information to the general-purpose environment in the form of a display ADU. These are sent to the general-purpose environment as a result of a command from the general-purpose environment or on a periodic basis. In the latter case, the ring mapper controls the display module such that it updates the general-purpose environment with a display ADU every update interval. Finally, display modules have schemes for handling flow control and race conditions between the general-purpose environment and the analysis and real-time environment.

Conclusion

The consultations with customer sponsors early in the development stages and throughout beta testing have given us a high degree of confidence in the usability of the FDDI Ring Manager. In addition to implementing the right features, we were able to address several real-life situations that would

not have occurred on the in-house test network. This collaboration with users has helped us create a product that answers the needs of our customers, the FDDI network managers and operators.

Acknowledgments

The FDDI Ring Manager was truly the result of a team effort. The contributions came from far beyond the R&D team including the marketing and marcom teams and our customer sponsors. We wish to thank Bill Barkley for developing the ring status functionality and providing the basic functions for parsing station management frames. We also wish to thank Frank Actis for asking his many "What if ...?" questions to ensure that the team was clear in its objectives with regard to the target customer. Thanks also to Steve Witt, the project manager, for giving us the freedom to do whatever was needed to realize the product. Thanks to Jim Pfaff and Mike Waughn for their marketing and support work and Aileen Johnson for coordinating other aspects of the product during its final stages. We also wish to acknowledge the contributions of a number of key individuals, too many to name here, who did intensive testing on the product in an effort to provide quality. Finally, many thanks to our customer sponsors who provided valuable input through all stages of the project.

Reference

1. *Hewlett-Packard Journal,* Vol. 43, no. 5, October 1992, pp. 6-47.
2. S. Bhat, "The Network Advisor Analysis and Real-Time Environment," *Hewlett-Packard Journal,* Vol. 43, no. 5, October 1992, pp. 29-33.

FDDI Topology Mapping

For the FDDI version of the HP Network Advisor protocol analyzer, ring mapping algorithms were devised to provide topological views of FDDI networks. These algorithms are designed to handle many problem situations that are characteristic of emerging LAN technologies.

by Sunil Bhat

Central to most network management products is network topology mapping functionality. A topology map helps in the management of the devices on the network and provides a context for troubleshooting network problems. The FDDI Ring Manager application for the HP Network Advisor protocol analyzer (see article, page 88) is no exception to this rule. The core of the FDDI Ring Manager is the ring mapper, which provides ring topology information that forms a convenient framework for gathering and maintaining connection, configuration, operational, and historical information for all devices on the ring. This article focuses on the processes and algorithms used by the ring mapper for generating topology maps for FDDI rings.

Topologically, an FDDI network forms a dual ring of trees or a subset thereof. This is because all FDDI links are duplex. This results in two separate counterrotating rings. In the event of a fault, the two rings wrap about the fault, forming a single unified ring. The dual ring is referred to as the *trunk* or *backbone* ring.

FDDI network topology can be viewed in two distinct ways. The *logical view* is an ordered sequence of all the active media access control (MAC) elements on the ring that participate in the FDDI protocol. It describes the path of the token through the active MACs. In a fault-free network, there are two separate token paths and therefore two logical rings. These are referred to as the *primary* and *secondary* rings. The *physical view*, on the other hand, describes the way in which the network components like stations, concentrators, and so on are attached to each other with physical links to form the FDDI network.

Basic FDDI Concepts

For the benefit of readers not familiar with the FDDI technology, this section provides a brief introduction to the basic concepts that are relevant to the the discussion of topology mapping. Fig. 1 shows a typical FDDI ring. The token path within stations is indicated using dotted lines.

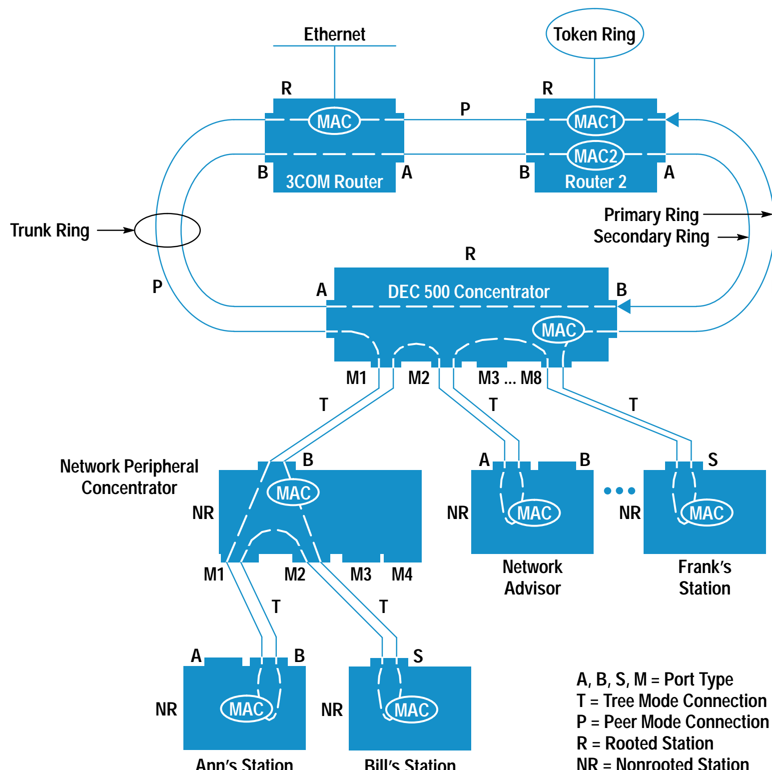


Fig. 1. An example of an FDDI ring.

A node on an FDDI network connects to the media via a *port,* which provides one end of a connection with another node. There are four types of ports: A, B, S, and M. Each connection is supported by a link which consists of a full-duplex fiber or copper cable. It has a mode associated with it based on the types of ports it connects. The valid modes are peer, tree, and none. Peer connections are formed on the trunk ring, which consists of A-B connections. These connections have no end port of type M. Tree connections connect nodes to the M-ports of concentrators. In this mode exactly one end of the connection is a port of type M. M-M connections fall under the none category and are disallowed.

The stations on the trunk ring are called *rooted* while those that connect to it via a concentrator are *nonrooted.* Rooted stations have no active port of type A, B, or S connected in tree mode.

FDDI differs from its preceding technologies like Ethernet and token ring in that the standard mandates that each node support station management (SMT). Station management is responsible for network reliability as well as management. It incorporates a management information base (MIB) that can be queried remotely using station management frame services to get information about the node, such as status, configuration, operation, and so on. This information is invaluable for understanding the topology and the health of the ring. The station management frame services that are used by the ring mapper are neighbor information frames (NIFs) and station information frames (SIFs).

Neighbor information frames are used in the neighbor notification process by every MAC on the ring. Each MAC generates a NIF at least once every 30 seconds, which allows others to monitor topological changes on the ring. A NIF sourced by a MAC contains its upstream neighbor address. This way each MAC can update its upstream neighbor address. NIFs include a description of the station including the type of node, number of MACs, port resources, the status of the station such as wrapped or twisted, and the upstream neighbor address of the logical upstream neighbor MAC.

Station information frames are used to acquire detailed station information in a packaged form. Configure and operation SIF responses contain configuration and operation information for each resource within the station, over and above the basic information contained in NIFs. This information is available only by active querying and includes basic station information contained in the NIFs, a path descriptor for the internal token path within the station, timer values and various frame counts for each MAC resource in the station, and link error information for each connection to the station.

Discovering the Ring Topology

There are both passive and proactive schemes for discovering the ring topology. The passive scheme consists of monitoring the asynchronous NIFs on the ring that are part of the neighbor notification process. A NIF contains quite basic information that lacks physical details, so one can only infer a logical view of the network. By monitoring the ring for a complete round of neighbor notification (at least 30 seconds), the mapper can obtain sufficient information to build the logical map. This map is representative of the ring (primary or secondary) being monitored.

Active schemes involve querying the nodes on the ring for neighbor and other information using NIF, SIF, or get-PMF request frames.† By using appropriate requests one can obtain logical, physical, configuration, and operational information. There are two flavors of active querying: broadcast and circular. In either scheme the responses are directed to the agent running the active process.

In broadcast querying the active process can broadcast one of the above request frames, capture all the responses, and build the map from the information contained in the responses. In circular querying the active process queries the nodes on the ring in a sequential manner by sending unicast request frames starting with its upstream neighbor. The neighbor's response to the query should provide the address of the next upstream MAC, which would be queried next. This process continues around the ring until the agent is reached. This process is also referred to as "walking the ring."

Data Structures

The ring mapper of the FDDI Ring Manager uses well-known structures to maintain information on each node in the network and its resources (MACs and ports). Each of these structures can be linked into a list. Sets of relevant structures are linked together in a specific order to represent logical or physical topological information. All of the structures and supporting services like linked lists, hashing services, and so on have been implemented in an object-oriented fashion using the C++ programming language.

The MAC structure contains the following information: this MAC address, upstream neighbor MAC address, downstream neighbor MAC address, and topological resource information. This information is maintained for physical topological mapping purposes only.

The port structure contains the following information: port type, connection state, remote port type, link to the remote port, and topological resource information. This information is maintained for physical topological mapping purposes only.

Topological resource information for a station relates to the token path within the station. The resources within a station are indexed from 1 to n+m, where n is the number of ports and m is the number of MACs physically present in the station. Both the MAC and the port resource structures also contain the following information:

- Resource index. The index of this particular resource among all resources in the station.
- Connection resource index. The resource index of the resource within the station that is the downstream neighbor of this resource in the token path.
- Neighboring resource information. This includes links to neighboring resource structures:
  - Pointer to the upstream resource
  - Pointer to the downstream resource.

The station structure stores basic information regarding a station and maintains links to each of its resource structures. This information includes a description including the type of the station and the number of MAC and port resources, the state of the station such as wrapped or twisted,

† A get-PMF request frame requests a parameter value from a MIB on the destination station. The response is a PMF—parameter management frame—containing the parameter value.

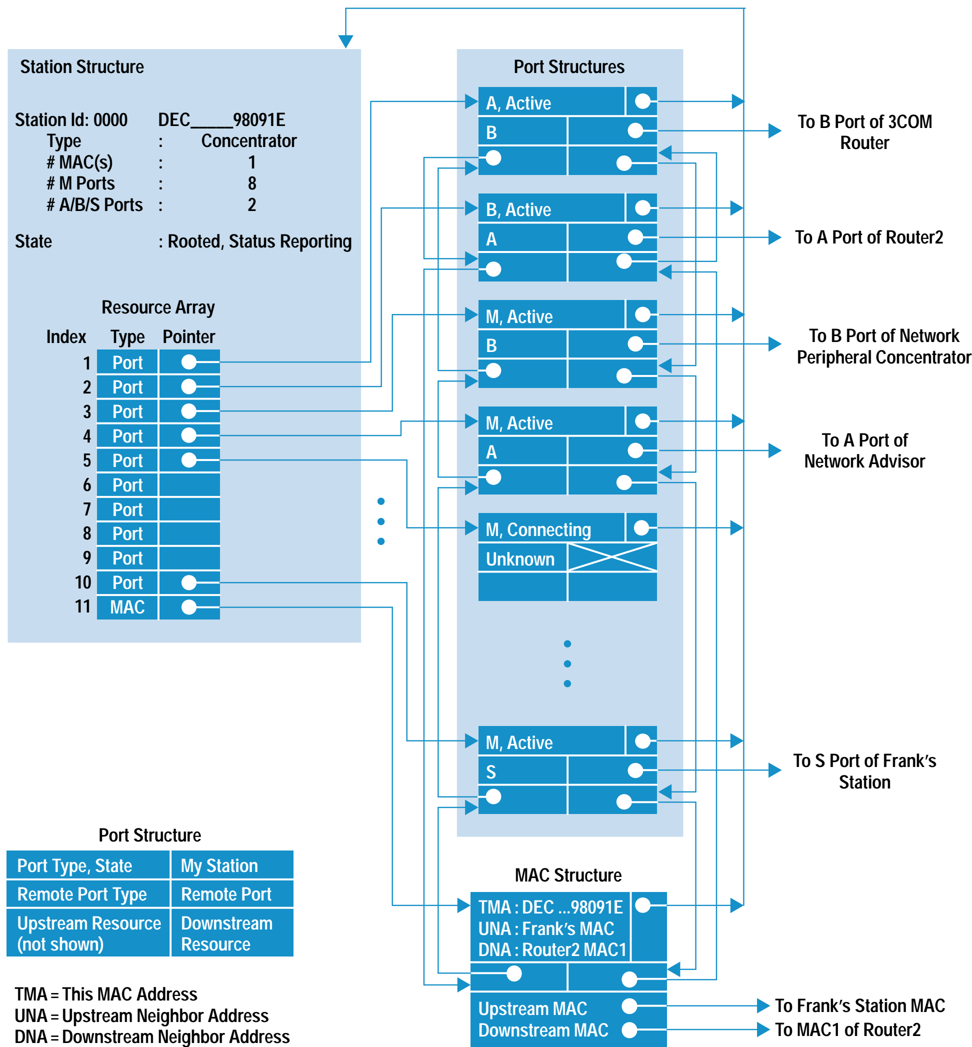


Fig. 2. The interconnection of a station and its resource structures for the DEC 500 concentrator of Fig. 1.

and an array of links to the structure for each resource in the station. Each link consists of a type of resource and a pointer to its structure.

An example of a station structure, along with its resource structures, is shown in Fig. 2. This structure is for a single-MAC dual-attach concentrator with eight M-ports. It also illustrates the internal token paths, which are represented by the upstream and downstream resource links.

Topological information regarding the FDDI ring is maintained by linking the above basic structures on specific lists. Each list provides different semantic information. There is at most one structure for any MAC. A MAC structure, once created, may exist on exactly one of the following three lists based on the current status of the MAC:

- The logical topology list consists of MAC structures, one for each active MAC on the ring in the reverse order of the token flow. In other words, for every MAC structure on the list, the structures to the right and left represent the upstream and downstream neighbor MACs.
- The active MACs list contains the structures for MACs that are active on the ring but have not yet been placed in their appropriate positions on the logical topology list.
- The removed MACs list represents the structures for the MACs that we know of but are currently inactive.

Any MAC structures that reside on any one of the above lists may also be directly accessed by means of a hashing scheme that keys off the least significant bits of its MAC address. The above scheme ensures that collisions in the hashing function are rare.

Timing Elements

The *monitor interval* is the interval of time over which the ring mapper monitors the relevant frames on the ring. The types of frames monitored depend on the scheme being used for topology mapping. For example, for logical mapping using the passive scheme, it suffices to monitor NIFs for an interval of 31 seconds to capture one complete round of the neighbor notification process. The *monitor timer* is used to track the monitor interval.

The *update interval* is the user-specified interval for determining the ring topology. In active schemes, this interval defines the polling frequency of the agent. This is tracked by the *update timer*.

Logical Mapping Process

The topology mapper can be configured to use either a passive or an active scheme to determine the logical view of the ring. This section describes only the passive process, which

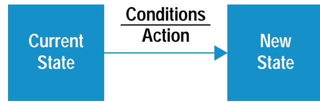


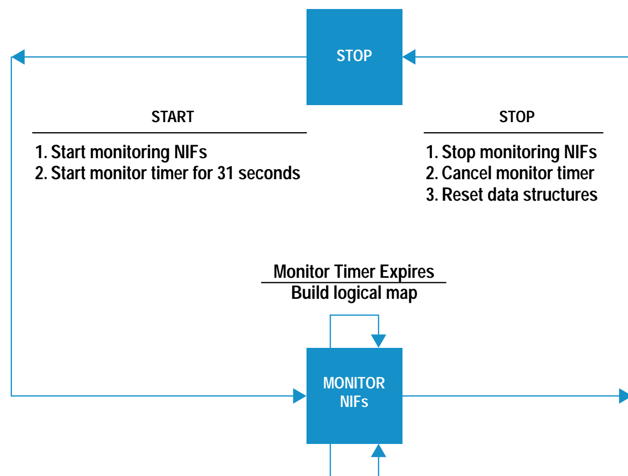
Fig. 3. In Figs. 4 and 6, all transitions from a current state to a new state are represented as shown here. All transition conditions are evaluated from the current state. If the transition conditions are satisfied, the state machine performs the associated transition actions and enters the new state. The new state now becomes the current state.

differs from the active process only in the frames (NIFs instead of SIF or get-PMF responses) it uses for obtaining neighbor information for building the logical topology.

The processes discussed in this article are described using state machines and the individual algorithms are described using simple procedural steps. The convention for the state machines is shown in Fig. 3.

Fig. 4 shows the state diagram for the passive logical mapping process. The mapper is in the STOP state when it is initialized (created). In this state, it waits for a user directive (START) to start the mapping process. On a START directive, it starts monitoring NIFs. For the purpose of knowing the upstream neighbor MAC address for each active MAC on the ring, it is sufficient to monitor only broadcast NIF requests and announcements for a minimum of 31 seconds.

In the MONITOR NIFs state the mapper is monitoring all broadcast NIF requests and announcements on the ring. On receipt of a frame from a sourcing MAC, it accesses (creates if necessary) and updates the structure for that MAC from the information in the frame. The receipt of a NIF from a MAC within a monitor period indicates that the MAC is active for



NIF Received (sa: Source Address, una: Upstream Neighbor Address)

```
1. active_MAC = find MAC structure for sa.
2. if (not found)
   then,
      2.1  active_MAC = create a MAC structure for sa.
      2.2  Append active_MAC to active MACs list.
   else
      2.3  if (active_MAC on removed MACs list)
           then,
                2.3.1 Remove active_MAC from removed MACs list and append to
                      active MACs list.
           endif
   endif
3. active_MAC.una ← una
```

Fig. 4. State diagram for the logical mapping process.

at least some of the monitor period. Therefore, if the structure for the MAC is on the removed MACs list it is moved to the active MACs list. The mapper exits out of this state when the monitor timer expires and it has finished processing all the received NIFs. At this point the mapper has all the necessary information to build the logical topology map. The mapper can be aborted out of this state by an explicit user STOP directive.

Building the Logical Map

Simply stated, the algorithm for building the logical ring map consists of starting with a MAC that was active (and thereby inserted) on the ring and finding the MAC structure for its upstream neighbor as reported in its upstream neighbor address. Once found it is placed upstream (to the right) of the current MAC. The algorithm then moves to the next MAC and repeats the process until it reaches the start MAC and the ring is complete. Complications to this process arise when there are inconsistencies or incompleteness in the ring information acquired over a monitor period. This causes discontinuities in the ring map which are referred to as *gaps* and are represented using a special MAC structure called a gap structure. Some of the causes for discontinuities are:

- A MAC with an unknown upstream neighbor address
- Multiple MACs reporting the same upstream neighbor address
- A MAC with an upstream neighbor address that is unseen
- Improper NIFs sent by a MAC.

If there are gaps a patching scheme is used to determine the ring map to the extent possible.

The basic steps of the logical mapping algorithm are described below:

1. Remove all MACs from the logical topology list that did not source a NIF over the last monitor period, that is, all inactive MACs. Invariant: A MAC is on the logical topology list or the active MACs list if and only if it sourced at least one NIF (i.e., was active) over the last monitor period.

2. Start with the last active MAC in the monitor period. This MAC has the best chance of being in the ring at the end of the monitor period. Let this be known as the start MAC. Set the current MAC to be the start MAC.

3. While (current MAC's upstream neighbor address differs from the start MAC's address)
   do
   3.1 Find the MAC structure for the MAC corresponding to the upstream neighbor address of the current MAC. This is the new upstream neighbor MAC of the current MAC. The old upstream neighbor MAC was the structure to the right of the current MAC.
   3.2 If ((current MAC's upstream neighbor address is unknown) OR
   (current MAC is one of multiple active MACs reporting same upstream neighbor address) OR
   (current MAC's upstream neighbor MAC has not sourced any NIF over last monitor period (i.e., is inactive)))
   then,
   3.2.1 Insert a gap structure to the right of the current MAC.

3.2.2 Cut the logical topology list past the gap node and append it to the active MACs list.
3.2.3 Patch logical map.
endif
3.3 If (new upstream neighbor MAC differs from the old upstream neighbor MAC)
then,
3.3.1 If (new upstream neighbor MAC is on the logical topology list)
then,
3.3.1.1 Remove MACs from (and including) old upstream neighbor MAC to (but not including) new upstream neighbor MAC and append to active MACs list.
else
3.3.1.2 Remove new upstream neighbor MAC from current list and insert to right of the current MAC.
endif
endif
3.4 Move to the next upstream MAC by setting the current MAC to be the new upstream neighbor MAC.
endwhile

4. Purge all gaps from the active MACs list. We will determine them afresh if needed.

5. If (active MACs list is nonempty)
then,
5.1 Insert a gap structure to the right of the current MAC in the logical topology list
5.2 Patch logical map
endif

6. Return SUCCESS

Patching the Logical Map

This functionality is invoked only if for any reason there is a gap in the logical map information. In the event of multiple gaps the map is broken into sequences such that the logical token flow information is consistent within a sequence. This algorithm uses a temporary list of MAC structures to build and store each sequence. Let us call it the sequence list.

1. Purge all gaps from the active MACs list. We will determine them afresh if needed.

2. While (active MACs list is nonempty)
do
2.1 Remove the MAC at the head of the active MACs list and append it to the sequence list. Let this be the current MAC.
2.2 Set sequence complete to FALSE
2.3 While (sequence complete is FALSE)
do
2.3.1 If ((current MAC's upstream neighbor address is unknown) OR
(current MAC's upstream neighbor MAC has not sourced a NIF over the last monitor period) OR
(current MAC's upstream neighbor MAC is on the logical topology list but is not the first MAC of a sequence))
then,
2.3.1.1 Insert a gap structure to the right of the current MAC.
2.3.1.2 Append entire sequence list to the logical topology list
2.3.1.3 Set sequence complete to TRUE
endif
2.3.2 If (current MAC's upstream neighbor is on the logical topology list and is the first of a sequence)
then,
2.3.2.1 Insert entire sequence list to the left of the upstream neighbor MAC.
2.3.2.2 Set sequence complete to TRUE
endif
2.3.3 If (current MAC's upstream neighbor MAC is on the active MACs list)
then,
2.3.3.1 Remove upstream neighbor MAC and append to sequence list
2.3.3.2 Move to next upstream MAC by setting the current MAC to be the upstream neighbor MAC
endif
endwhile
endwhile

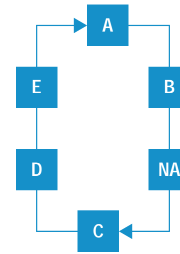


Fig. 5. The primary ring of an FDDI network.

Logical Mapping Example

Consider the ring in Fig. 5, where the labels on the nodes refer to their MAC addresses. The token flow is as indicated.

The HP Network Advisor protocol analyzer with the FDDI Ring Manager application is the station labeled NA between nodes B and C. Let the sequence of NIFs monitored by the Network Advisor in the first monitor interval (say 31 seconds) be as listed below in increasing order of time.

Source Address	Upstream Neighbor Address
D	C
NA	B
E	?? (unknown)
B	A
A	E

In this list there are two gaps in our information: (1) Node E reports an unknown upstream neighbor address, and (2) There is no NIF from node C (maybe because of a poor implementation) within the entire monitor interval, so we are unable to verify its existence on the ring.

Since the last NIF monitored within the monitor interval was from node A, we shall start building the logical map from it.

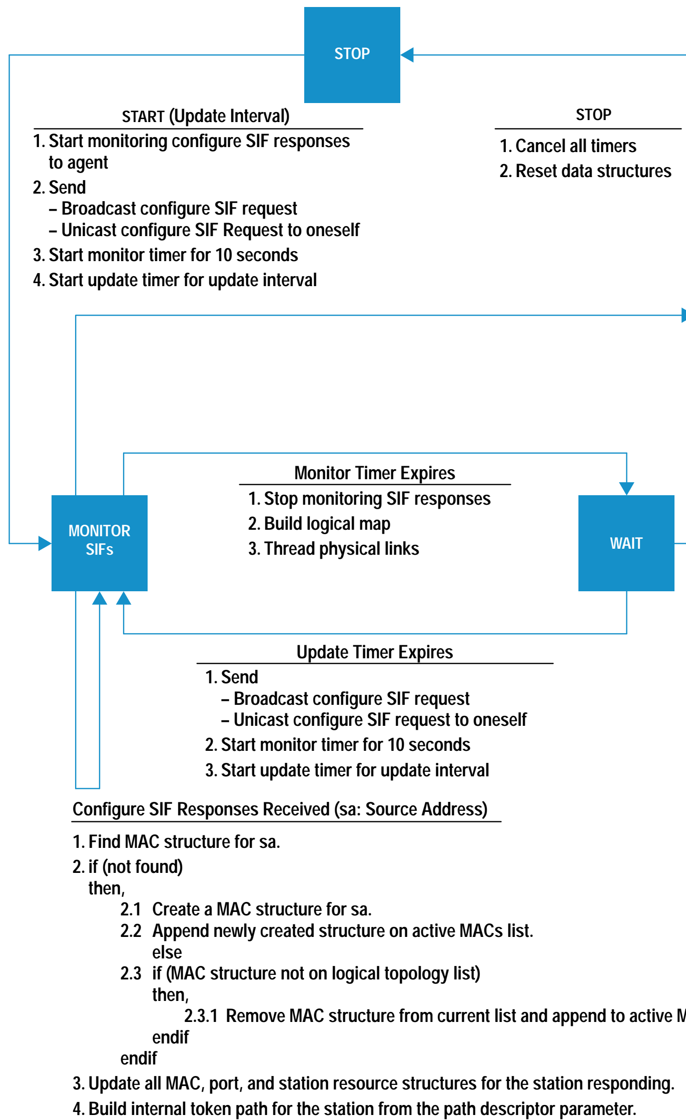
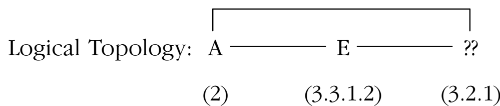
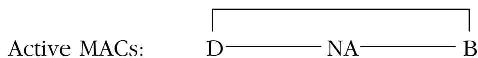


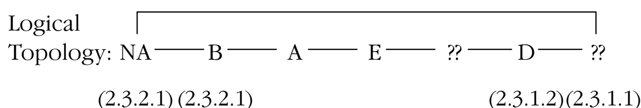
Fig. 6. State diagram for the physical mapping process.

On applying the build logical map algorithm we build a partial sequence before encountering the first gap (represented as ??) in our information. The configurations of the logical topology and active MACs lists at this point are shown below along with the step number of the logical mapping algorithm by which the MAC was added to the logical topology list. Since both lists are double-linked lists, the links are full-duplex.





At this point, the patching algorithm is invoked, which results in the following logical topology. In this case the step numbers reflect the steps of the patching algorithm.



Physical Mapping Process

To build a map of the physical topology of the network, the mapper needs logical as well as internal token path information for all of the active stations on the network. Logical information can be obtained in a passive or an active manner as outlined in prior sections. However, path information can only be obtained by actively querying for it.

The physical mapping process consists of three major steps:

- Obtain the internal token path for each active station on the ring.
- Build the logical topology map of the ring.
- Determine physical links between stations by threading end ports of physical connections using the logical map and the internal token paths of the end stations.

Fig. 6 shows the state diagram for the physical mapping algorithm. The STOP state is the same as for the logical mapping process. On a START directive from a user, the monitor timer is set to an interval (10 seconds) that should be sufficient to receive all responses to the requests. The update monitor is set for a user-defined interval, which needs to be greater than the monitor period.

In the MONITOR SIFs state the mapper is monitoring configure SIF responses destined for it. On receipt of a response, it accesses the station structure for the sourcing MAC and updates the MAC and port resource structures for all resources within it. It transitions to the WAIT state when the monitor timer expires. At this stage it is assumed that the mapper has received all the responses to its requests. The mapper now has all the information to build the logical and physical topology maps. The mapper can be aborted out of this state by a STOP directive.

In the WAIT state the mapper is waiting for the expiration of the update interval at which time it will query the nodes again and transition to the MONITOR SIFs state. The mapper can be aborted out of this state by a STOP directive.

The token flow through the resources of a station, both MACs and ports, is described by the path descriptor parameter of the station management MIB of the station. This information can be requested using configure SIFs or get-PMF request frames. The path descriptor is a mandatory parameter in the configure SIF response. A Network Advisor decode of the path descriptor for a concentrator node is shown in Fig. 7. It contains a record for each of the resources that are physically present in the station, first the ports and then the MACs.

All resources physically present in a station are labeled using sequential indexes. The ports have indexes from 1 to n, where n is the number of ports physically present in the station. The MAC resources are indexed from n+1 to n+m where m is the number of MACs physically present in the station. For a resource in the token path, the connection resource index in its record indicates the index of its downstream neighbor resource within the station. For a resource that is not part of the token path, the connection resource index is its own index. To determine the internal token path, choose a resource that has a connection resource index different from its index. Move downstream to the next neighbor resource by following its connection resource index. By following the connection indexes of the resources in this fashion we can traverse the internal token path, setting both upstream and downstream neighbor resource links appropriately in the process. For the concentrator of Fig. 1 the internal token path and the resource links that reflect it are shown in Fig. 2.

Threading Physical Links

The task of determining the physical links consists of threading through the token path of the entire ring starting from a rooted MAC and moving upstream through the resources on the token path. During this traversal, the algorithm links remote port resources together to reflect actual physical links between these ports. After a successful completion of the algorithm, it is possible to identify the actual physical topology by following the resource and remote port links of the active resources on the ring. The logical topology list provides a basis for the threading process. It represents all active MACs on the ring in the reverse order of token flow. The threading algorithm is as follows:

1. Clear the remote port link for all ports of all stations that we know of. These links will be established afresh as part of the threading process.

```
Frame: 27        Time: Mar 21@ 9:10:59.7015576  Length: 197
                        ***** DETAILED FORMAT *****
Frame Class             2             SIF Configuration
Frame Type              3             Response
Version Id              1             Version 6.2
Transaction Id          303174162
Station Id (User)       0000          User defined section
Station Id (IEEE)       DEC---98091E  IEEE defined section
Header Pad              0
Info Field Length       160           Total length of station management
parameters present
                        .
                        .
                        .
Parameter               00–08         Path Descriptor
 Length                 88            Remaining parm octets
 Pad/MIB Index (7.2)    0
 Port Type              0             Type A
 Connect State          3             Active
 Remote Port Type       1             Type B
 Remote MAC             1             Remote MAC Indicated
 Con Resource Index     3             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              1             Type B
 Connect State          3             Active
 Remote Port Type       0             Type A
 Remote MAC             0             None present
 Con Resource Index     1             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          3             Active
 Remote Port Type       1             Type B
 Remote MAC             1             Remote MAC Indicated
 Con Resource Index     4             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          3             Active
 Remote Port Type       0             Type A
 Remote MAC             1             Remote MAC Indicated
 Con Resource Index     10            Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          1             Connecting
 Remote Port Type       4             Unknown
 Remote MAC             0             None present
 Con Resource Index     5             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          1             Connecting
 Remote Port Type       4             Unknown
 Remote MAC             0             None present
 Con Resource Index     6             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          1             Connecting
 Remote Port Type       4             Unknown
 Remote MAC             0             None present
 Con Resource Index     7             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          1             Connecting
 Remote Port Type       4             Unknown
 Remote MAC             0             None Present
 Con Resource Index     8             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          1             Connecting
 Remote Port Type       4             Unknown
 Remote MAC             0             None Present
 Con Resource Index     9             Port Connection Index
 Pad/MIB Index (7.2)    0
 Port Type              3             Type M
 Connect State          3             Active
 Remote Port Type       2             Type S
 Remote MAC             1             Remote MAC Indicated
 Con Resource Index     11            Port Connection Index
 MAC Address            DEC---98091E
 Con Resource Index     2             Mac Connection Index
----End----
```

Fig. 7. Network Advisor decode of the path descriptor parameter in an SIF configuration response.

2. Set the start MAC to be any rooted MAC from the logical topology list. (This choice of the start MAC works fine only if the ring is neither wrapped nor twisted. Those cases are discussed in the next section.) Set the current MAC to be the start MAC.

3. Set the end MAC to be the first MAC downstream of the start MAC that belongs to a different station than the start MAC. The end MAC can easily be found by traversing the logical topology list in the downstream direction and comparing the station ID for the host station of each traversed MAC with the start MAC.

4. While (current MAC different from end MAC)
   do
   - 4.1 Find the first MAC upstream of the current MAC on the logical topology list that belongs to a different station than the current MAC. Let this be the next MAC.
     After step 4.1 we have two MACs that belong to neighboring stations. The next step is to find the end ports for the physical connection between these stations.
   - 4.2 Traverse the internal token path from the next MAC in the downstream direction until a port resource is reached. Let this be the next port.
   - 4.3 Traverse the internal token path from the current MAC in the upstream direction until a port resource is reached. Let this be the current port.
   - 4.4 While (current port is already linked to another remote port)
     do
     - 4.4.1 Current port = current port's remote port
     - 4.4.2 If (current port is same as next port)
       return SUCCESS
       This happens in the case of a wrapped station.
     - 4.4.3 Traverse the internal token path from the current port in the upstream direction until a port resource is reached

     endwhile
     At this point we have two end ports of a physical connection. Set the port structure links to reflect this derived information.
   - 4.5 Link the remote port links together:
     remote port of current port = next port
     remote port of next port = current port
   - 4.6 Move to the next MAC. It becomes the new current MAC.

   endwhile

5. Return SUCCESS

Problem Situations

The algorithms discussed so far in this article describe the basic framework for generating logical and physical views of FDDI ring topology. They work well for rings having conforming nodes attached to them. Real FDDI installations pose special challenges to the above schemes, especially the one for building the physical maps. By the very nature of the map, a lot of information from all of the active nodes on the ring, which may be from different vendors, needs to fit together somewhat like a jigsaw puzzle to get a consistent picture of the physical links. For a variety of reasons it may not always fit.

The basic threading algorithm as described above works well if for any two neighboring MACs on the logical map that belong to separate physical nodes, there exists a physical link connecting these nodes. Let this be known as the threading condition. Most problems or complications of the threading process arise when for any reason this condition is violated. It is therefore important to detect all such violations and handle them appropriately. We made significant enhancements to the basic structure of the above algorithms to deal with a variety of such situations, some anticipated and others not. These situations can be broadly classified as topological variations, MACless nodes, incomplete information, and inconsistent information.

Topological Variations. This category of problem situations includes different ring topologies such as a twisted ring, a wrapped dual ring, and a wrapped and twisted ring.

In the event of a twisted ring that is not wrapped, some nodes on the ring disappear from the primary ring and move to the secondary ring such that there is loss of communication between the two sets of nodes. Even though all of the nodes are still physically connected to each other in the form of a ring, the HP Network Advisor will see only a subset of the nodes based on the ring (primary or secondary) it is monitoring. For example, Fig. 8a shows a twisted ring that

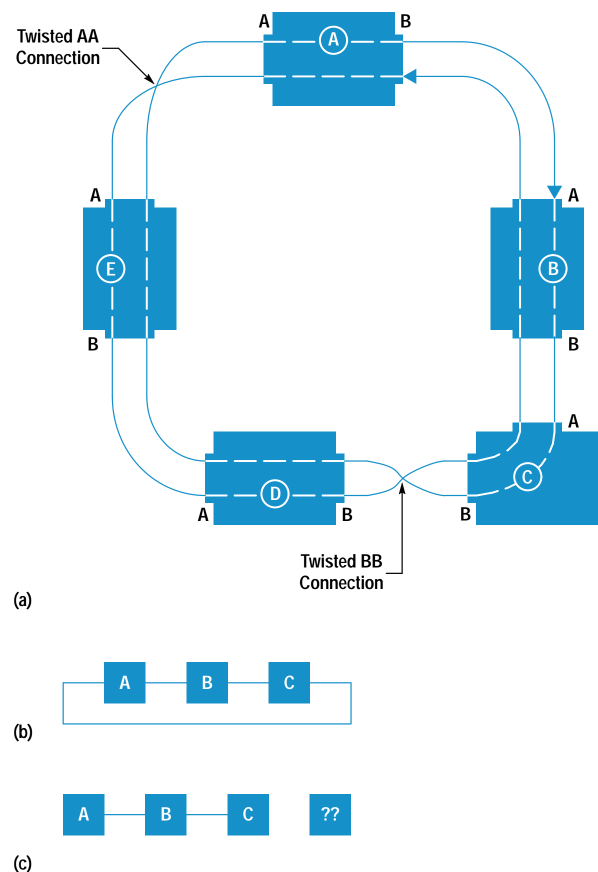


Fig. 8. (a) A twisted dual ring. (b) The output of the basic threading algorithm for the ring in (a). (c) The more accurate physical map generated by the modified threading algorithm.

consists of two disjoint logical rings—A-B-C and E-D, in the order of token flow. If the basic threading process used the first logical ring to determine the physical map, then the result would be as shown in Fig. 8b, which is inaccurate since it shows a direct physical link between nodes C and A, the points of the twist. The ring mapper of the FDDI Ring Manager application of the HP Network Advisor handles this by checking for neighboring stations on the ring that are twisted (it does this by looking at port types) and inserting a gap between them. This results in the map of Fig. 8c, which is as accurate as we can get given the information that we possess.

A wrap on the dual ring causes the primary and secondary rings to coalesce to form one active ring. A wrapped dual ring is handled by choosing the start MAC to be the MAC in the station that is wrapped on its port A and the end MAC to be the first MAC downstream of the start MAC that belongs to a different station. Since the threading process moves upstream, this choice ensures that the traversal of the logical map from the start MAC to the end MAC will not violate the threading condition.

If the wrapped nodes happen to be twisted as well, then it is difficult to use the solution just described for the wrapped dual ring. Typically rings have most if not all of their MACs on the primary ring. Given this knowledge, we need to choose a start MAC that follows the threading condition, that is, there is a physical link to the next upstream node. This can be done by traversing the logical map in the upstream direction from each wrapped MAC to the other, counting the MACs in between. The start MAC chosen is the one that gives the higher count.

MACless Nodes. These nodes pose an especially difficult problem in the sense that they are difficult to detect and even more so to handle. They are invisible to the logical mapping process and are therefore not reflected in the logical map. This causes problems while threading the physical links especially if the MACless node is a rooted concentrator.

For the sake of simplicity, the ring mapper does not handle MACless nodes on the backbone ring. It uses heuristics to place MACless nodes that are connected to a concentrator. It checks the remote port type for each active M-port of the concentrator and tries to match it with the port type of all nodes with one or more MACs connected to them. A mismatch may be an indication of the existence of a MACless device connected to the concentrator. It may also be a result of improper information reported by the concentrator.

Incomplete Information. Lack of responses from any number of active nodes will create a gap in our knowledge about the ring, causing problems in the threading process. These gaps are represented as such in the map. A gap in the map extends from the start of the discontinuity to the next rooted node.

Inconsistent Information. While FDDI is well-defined as a standard, it is still maturing as an implemented LAN technology and there are a large number of semiconforming implementations of station management currently on the market. As a protocol analyzer vendor we need to be able to deal with numerous versions of FDDI products from numerous vendors. The FDDI Ring Manager application is one of the first of its kind that actively queries for management information and pieces it together. It is also the first to deal with the myriad of problems associated with improper responses. Our experience has been that the information in NIFs is more stable and reliable than those in SIF responses. Therefore, the ring mapper's active mode of operation was extended to make monitoring of NIFs optional. However, this does not eliminate inconsistencies, and these are handled much like incomplete information.

The above problem situations get worse when several of them are present on the same ring. As discussed above, the physical mapping process tries to isolate the domain of the problem and uses gaps to signify discontinuities in ring information. Since the physical threading process starts at a MAC and moves upstream, a problem situation at any point in the threading process may cause a gap in information from the current node to the next upstream rooted node.

Conclusion

The topology mapping algorithms discussed in this article provide valuable topology information to the network administrator operator. Being one of the early manufacturers to provide physical mapping capability, we had to deal with a number of problems that stemmed from incomplete implementations of station management. By using the Interoperability Laboratory at the University of New Hampshire and working closely with customers early in the project, we obtained a real understanding of FDDI networks that enabled us to enhance our algorithms to deal with problem situations. This allowed us to deliver a product that our customers could use under adverse conditions. This is viewed as real value by our customers who feel that they have an increased visibility to their FDDI networks.

Acknowledgments

First and foremost, I would like to thank Niel McKee, then working for HP Laboratories Bristol, whose early prototype work with regard to FDDI topology mapping algorithms jump-started this project. Thanks also to Bill Barkley, who designed the FDDI ring status application and the NIF and SIF frame arrival portion of the ring mapper. Many thanks to Ron Pashby and his team at the Interoperability Laboratory at the University of New Hampshire for their feedback and testing efforts. Special thanks to Murali Krishnan, who helped me enormously in isolating and characterizing problems with the software to ensure quality. I also acknowledge the enormous testing effort by all the folks involved. Finally, thanks to Steve Witt who made it all possible.

Automation of Electrical Overstress Characterization for Semiconductor Devices

An automatic test system has been developed to characterize semiconductor devices and interconnect failures caused by electrical overstress (EOS). Electrical stress in the form of current pulses of increasing amplitude is applied to a device until it reaches a prespecified failure criterion. The system was developed for monitoring EOS robustness in advanced CMOS processes.

by Carlos H. Diaz

Semiconductor devices have a limited ability to withstand electrical overstress (EOS). Device susceptibility to EOS increases as the device is scaled down to submicrometer feature sizes. At present, EOS is one of the major causes of IC failures.[1,2,3] EOS embodies a broad category of electrical threats to semiconductor devices, including electromagnetic pulses (EMP), electrostatic discharge (ESD), system transients, and lightning. However, common use of the terminology puts ESD in a separate category so that EOS means electrical overstress other than ESD. We will follow this convention in this article.

Electrostatic discharge occurs whenever a charged object is grounded, resulting in the release and equalization of the static charge. ESD events are generally in the submicrosecond domain. They may occur any time an IC chip is touched by human hands, held with metal tweezers, or contacted by a grounded metal object such as when the devices slide down plastic tubes in a test engineering environment and an IC's corner pins come into contact with the grounded metal rails.

EOS events, on the other hand, may last several microseconds or even milliseconds and are commonly associated with overvoltage and transient spikes under IC test conditions or in IC applications such as in system boards.

At the chip level, ESD damage can cause increased leakage at the I/O pins, increased standby current, or in extreme cases, full circuit failure. Failures can occur at the board and system levels as well. IC pin failures can range from very small melted filaments at the junctions to gross damage at the pin site.

The physical effects of ESD and EOS on ICs can be categorized as thermally induced or electric field induced failures. Among the thermally induced failure mechanisms are drain junction damage with melted filaments, polysilicon gate filaments, contact metal burnout, and fused metallization. Typical field induced ESD-related failure mechanisms are dielectric breakdown (gate oxide rupture) and latent hot-carrier damage.

Usually, the damage signature indicates whether a failure was ESD-related or EOS-related. In general, ESD-related damage is associated with small failure sites—for example, oxide pinholes, polysilicon filaments, or junction deformation occurring preferentially at diffusion corners.

EOS damage, on the other hand, can be quite drastic and cover most of the area around the pin. This kind of damage is commonly caused by long-lasting (>1 μs) conditions for which the device failure (generally associated with thermal runaway) happens early during the stress event. Thus, what was probably a small failure site just after the onset of thermal runaway invariably grows in size as a result of the additional energy delivered to the structure during the rest of the stress event.

Regardless of whether stress events are ESD-related or EOS-related, the failure sites are in general confined to the protection circuits. However, for certain types of ESD and EOS events, the protection devices or the coordination among the protection structures within an IC may prove to be ineffective, in which case failure sites will also be found in the internal circuitry.

ESD Test Methods

Currently, the most commonly used models for describing various categories of ESD pulses that affect ICs during handling are the human body model, the machine model, and the charged-device model.[4,5,6] The basic model for ESD protection is the human body model (HBM), intended to represent the ESD caused by human handling of ICs. The HBM equivalent circuit is shown in Fig. 1a. The 100-pF capacitor is charged with a high-voltage supply and then discharged through a 1.5-kilohm resistor onto the pin under test. ESD-HBM is the most widely used method of qualifying the ESD performance of on-chip protection circuits and is standardized.[4] Typically, HBM events occur at 2 to 4 kilovolts in the field, so protection levels in this range are necessary.

Contact with machines is also an ESD-type stress event. The ESD machine model (MM) is intended to model the ESD

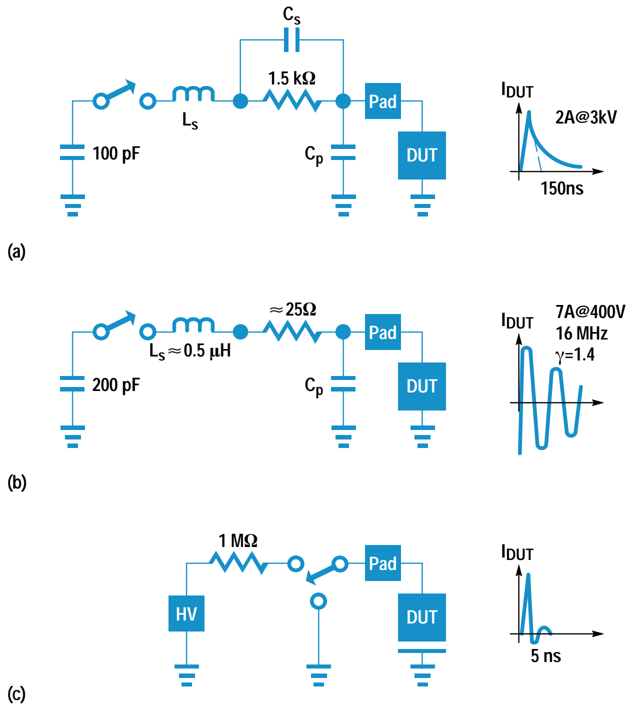


Fig. 1. Equivalent circuits of electrostatic discharge (ESD) models. (a) Human body model (ESD-HBM). (b) Machine model (ESD-MM). (c) Charged-device model (ESD-CDM).

produced by a charged object making contact with ICs during device bonding, assembly, or testing. ESD-MM testers deliver damped, oscillating, 16-MHz stress currents (first peak value in the order of 7A when the 200-pF capacitor is precharged to 400V) to the device under test.[5] A schematic representation of an ESD-MM stress tester is shown in Fig. 1b. In contrast to the ESD-HBM test method, there is no unique definition of the ESD-MM method.

The ESD charged-device model (CDM) (Fig. 1c) is intended to model the discharge of a packaged IC. Charges can be placed on an IC either during the assembly process or on the shipping tubes.[6] ESD-CDM testers electrically charge the device under test (DUT) and then discharge it to ground, thus providing a high-current, short-duration (5-ns) pulse to the device under test. As in the ESD-MM case, there is no industry agreement on the ESD-CDM test method specifications.

EOS Test

The EOS test is more complicated because of the wide spectrum of electrical characteristics of the stress events to which an IC may be subjected in its lifetime.[7] Currently, there exist no EOS standards or quantitative EOS design objectives, thus limiting or delaying the designer's attention to the EOS problem. Constant-current pulses are commonly used for EOS testing.[2,7,8] Such EOS stressing is easy to generate consistently and is also amenable to simple analysis. For thermally induced failures, failure thresholds for any stress waveform can, in principle, be obtained from the failure threshold derived under pulse-stress conditions and given in terms of the relationship of current-to-failure I_f and time-to-failure t_f. For example, constant-current pulses lasting 100 to 250 ns are used to study second-breakdown phenomena in NMOS devices. The failure current levels measured using these short pulses are treated as predictors of the HBM-ESD failure thresholds.[8]

NMOS Transistor

The most commonly investigated and well-understood ESD phenomenon is the ESD behavior of an NMOS transistor. Consider the operation of an NMOS device under high-current conditions. The basic I-V characteristics are shown in Fig. 2 for the gate and substrate tied to ground. Here V_{t1} is the drain junction breakdown voltage. Holes from the drain impact ionization process are injected into the substrate, increasing the substrate potential near the source junction. When enough hole injection is present, the source junction becomes forward-biased and the parasitic n-p-n bipolar transistor enters active mode and causes the snapback phenomenon. During an ESD or EOS event, the device operates primarily in the snapback mode. The device terminal voltage is determined by the snapback voltage V_{sp}, the contact resistances, and the level of device self-heating. At high stress levels, the device could go into second breakdown, the region where the device temperature has increased to such a level that thermal carrier generation is high enough to dominate the conduction process. Second breakdown is a positive feedback process that causes device failure because of current localization. The current level at which the device undergoes second breakdown (I_{t2}) is used as a predictor of the device's current-handling capabilities under ESD events.

The test system introduced here is designed to determine this kind of transient I-V characteristic for devices at both the wafer and package levels. It is an EOS tester, that is, it applies constant-current pulses to the DUT. The ESD performance of the DUT is inferred from the EOS test results. The advantage of this approach is that the EOS test can be applied at the wafer level while direct ESD tests cannot be applied at the wafer level.

Test System Description

In the EOS test system, stress current pulses are generated using a high-power pulse generator or a transmission line system. The HP-IB-programmable HP 8114A pulse generator is used to deliver current pulses of up to 1A amplitude and 50-ns-to-1-s pulse width. For submicrosecond pulses higher than 1A, a coaxial cable charged to a high voltage is used as the stress source.[9] The length of the coaxial cable determines the pulse width at the device under test. To avoid reflections, the line is terminated by a 50-ohm resistor at the near end.

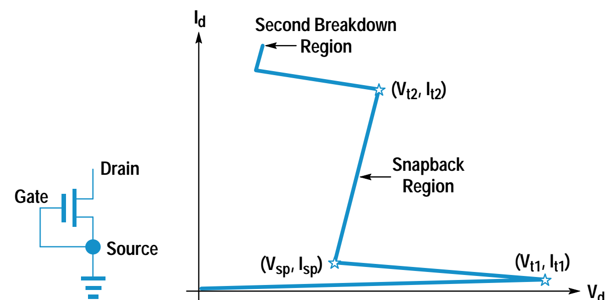


Fig. 2. Schematic I-V characteristics of a grounded-gate NMOS transistor.

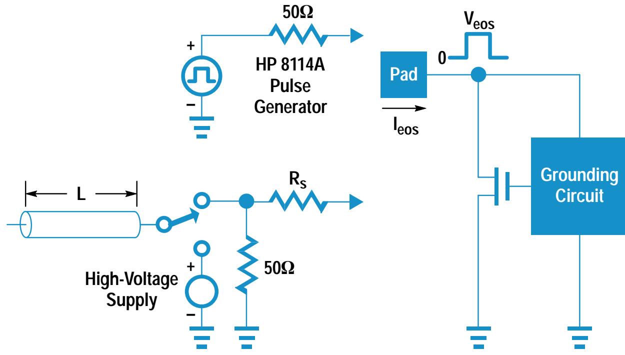


Fig. 3. Basic stress source configuration. The source is an HP 8114A pulse generator for current pulses up to 2A and a coaxial cable charged to a high voltage for pulses greater than 2A.

Additionally, to eliminate the effects of DUT impedance variations and to increase stability in the negative differential region, a 100-to-500-ohm series resistance is used. The basic stress setup is shown in Fig. 3.

Fig. 4 is a schematic of the automated test system. The main components are the stress sources described above, a real-time high-speed digital storage oscilloscope (HP 54720A), a semiconductor parameter analyzer (HP 4142A), a switch matrix control unit (HP 75000 VXI mainframe with a Form-C switch board), and a workstation running the HP IC-CAP circuit and device modeling software package.

New IC-CAP instrument drivers, written in C++ using the IC-CAP open measurement interface, were developed for the HP 8114A pulse generator, the Bertan 225 high-voltage power supply, and the HP 54720A digital oscilloscope. The HP 75000 VXI mainframe and its switch board cannot be made IC-CAP instruments because of their special addressing protocol. To overcome this limitation, C routines were written to perform basic interface and control functions. These C routines are available to the instrument drivers, and are also available as command line functions. Fig. 5 illustrates the IC-CAP instrument option tables for these instruments. These tables provide the user with some of the most frequently used instrument features. For example, the user can choose either real-time or equivalent time acquisition mode and the sampling rate of the HP 54720A. For the HP 8114A pulse generator, the pulse count, trigger mode, output impedance, and other features can be specified. The option tables for the pulse generator and the high-voltage power supply contain fields that specify the base addresses of the switches that isolate the source from the test system (if the system is configured without the VXI mainframe, these instruments can still be used by setting the primary address of the switch fields to an illegal value such as 32).

Fig. 6 shows the IC-CAP window for a typical pulse stress measurement. Once the IC-CAP model variables for the pulse amplitude and width and the time-sweep window are defined, the execution of the measurement command on the IC-CAP pull-down menu causes the device to be stressed and the corresponding current and voltage waveforms to be recorded. Fig. 7 shows such waveforms for a low-voltage silicon controlled rectifier used as on-chip ESD protection in a submicrometer CMOS process. After stressing, the user can

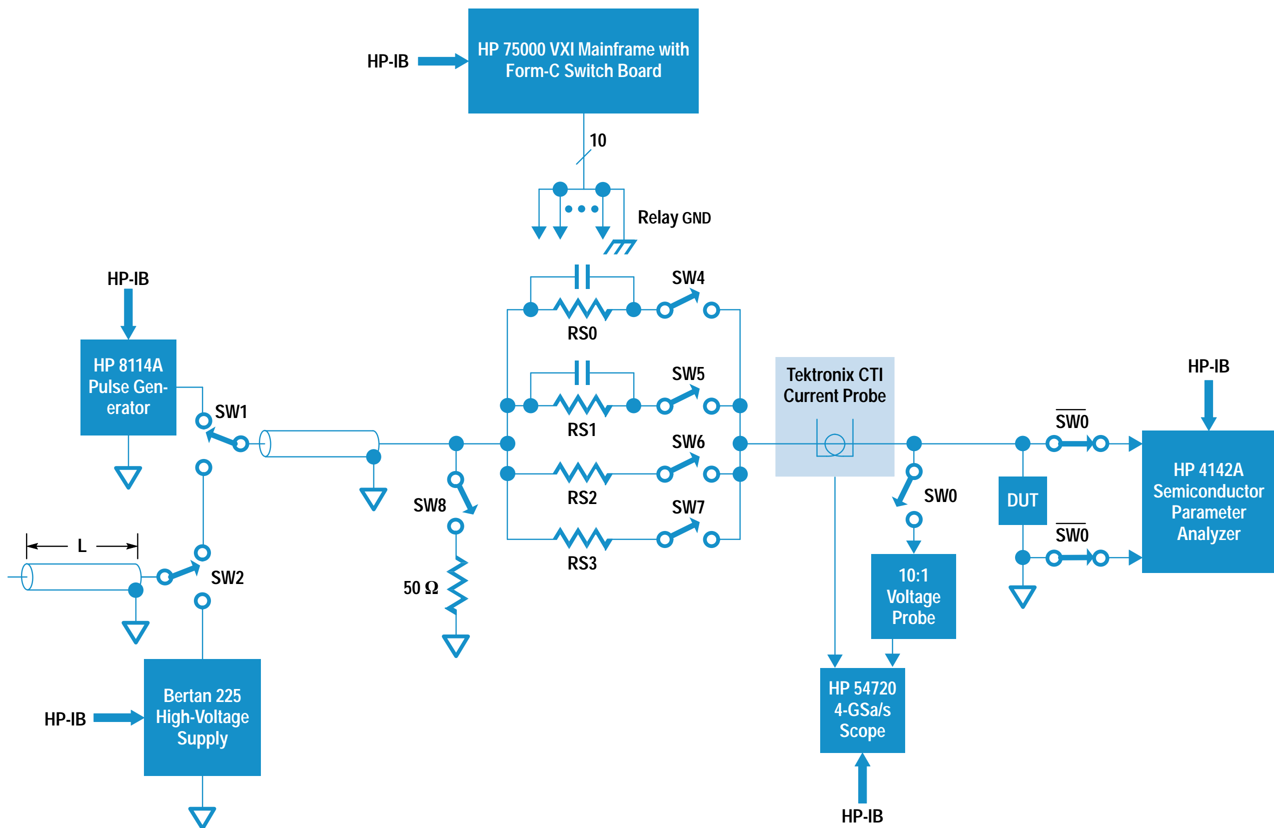


Fig. 4. Schematic diagram of the IC-CAP-based EOS test system.

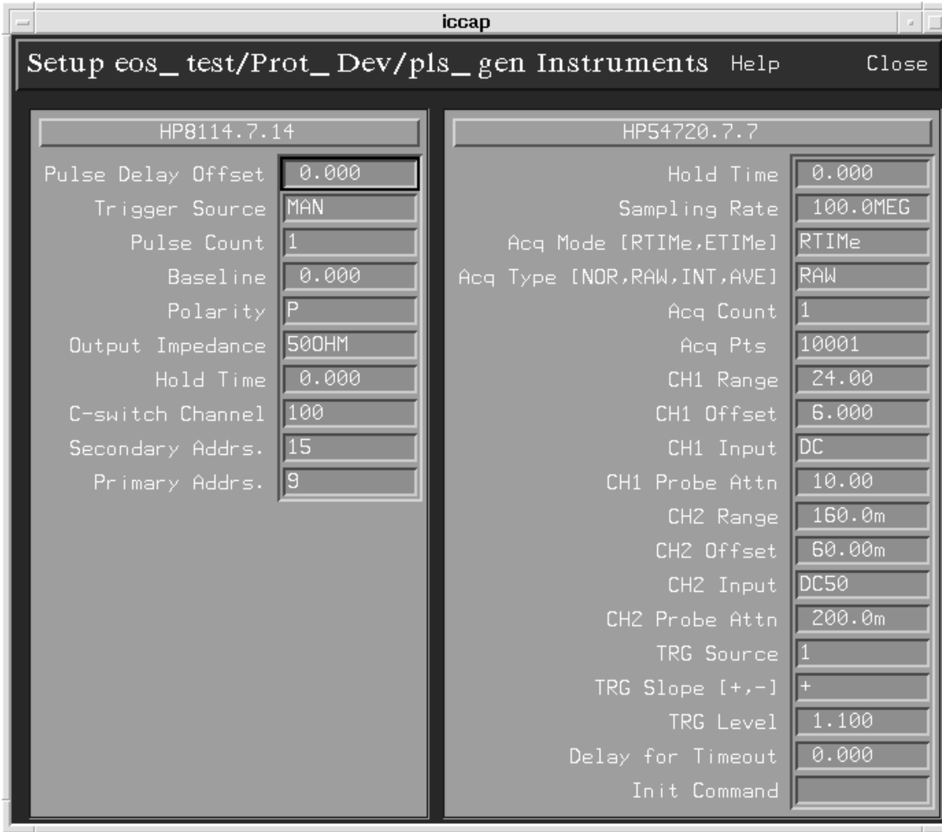


Fig. 5. IC-CAP instrument option tables for the pulse generator and oscilloscope.

perform HP 4142A measurements and compare the results with prestress data to determine if the device has been degraded by the stress event. Test automation to evaluate device failure thresholds is then a simple task of interleaving stress and device parameter measurements in a repetitive fashion.

Test programs are written as IC-CAP macros for characterization of second-breakdown phenomena in semiconductor devices and for pulsed electromigration of interconnect lines in IC processes. Another test macro was written to measure time-dependent dielectric breakdown (TDDB) for CMOS

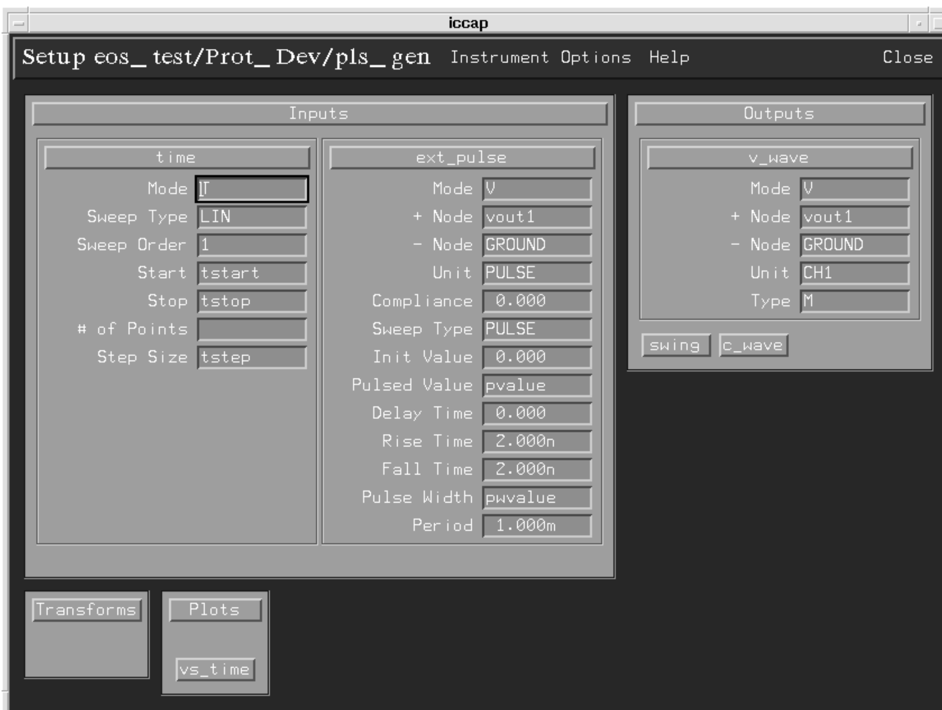


Fig. 6. IC-CAP window for a typical pulse stress measurement.

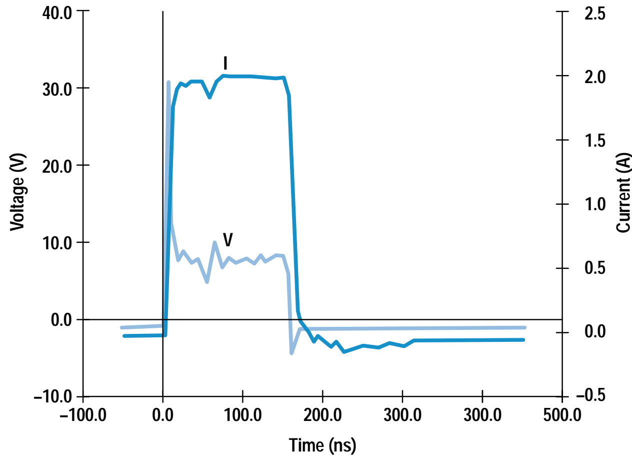


Fig. 7. Current and voltage waveforms recorded in a stress test of a low-voltage silicon controlled rectifier used as on-chip ESD protection in a CMOS process.

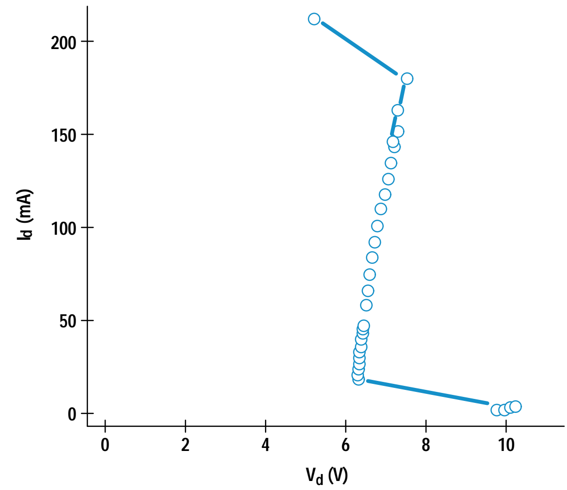


Fig. 9. Pulsed I-V characteristics of a submicrometer NMOS transistor.

gate oxides. These IC-CAP macros determine the test conditions and DUT information, coordinate the stress and measurement sequences, and generate the test reports. A typical report generated by a test macro for a particular batch is shown in Fig. 8. Each line in the device data is the result of a measurement taken after delivering a single pulse to the device and subsequent measurement of the device leakage as a failure condition. From this data, the device transient I-V characteristic can be reconstructed. This is shown in Fig. 9 for an NMOS device in a submicrometer technology. Test files containing raw test data, like the one shown in Fig. 8, can be postprocessed and fed into a database and data analysis package such as S-Plus. Device electrothermal model parameters can be extracted in IC-CAP and fed to circuit-level electrothermal simulators such as iETSIM[10] that are capable of modeling device behavior up to the onset of second breakdown.

Another IC-CAP test macro was developed to determine the maximum pulsed current density in metal lines for an advanced CMOS process. Fig. 10 presents the results of this characterization for the three-level-metal system as a function of the stress pulse width.

```
08/04/94     11:27:59
Mask:    1th7b
Lot:     yth15
Wafer:   9
Test:    eos
Ioff Spec =      100e−9 @ Vdut =      4 Volts
2ndBV Spec =     −1
Pulse width =    1.5E−07

        .
        .
        .

Device cgnch8f
Die 84 Reference leakage current If_rel = 4.812E−11
Vdut_pk  Vdut [V]  Idut [A]  I_off [A]
9.78   9.7826  0.0029    1.18E−11
9.79   7.1804  0.016242  8.138E−11
9.9    6.5744  0.027964  7E−13
10.01  2.957   0.045094  5.0856E−08
9.84   6.0052  0.048454  5.0364E−08
9.34   6.034   0.058052  4.8872E−08
9.01   5.986   0.067442  4.8164E−08
9.22   6.0008  0.077012  4.7708E−08
9.89   6.0238  0.13716   4.7164E−08
11.39  6.0264  0.21542   4.633E−08
12.71  2.6006  0.32224   9.9996E−07
Failed ISpec: Idut = 0.21542, Ioff = 9.9996E−07
        .
        .
        .
```

Fig. 8. Typical report generated by the test macro for a particular batch of devices under test.

Conclusion

An automated test system has been developed for the pulse characterization of semiconductor devices and interconnect metal lines in MOS processes. The system is built on a test environment based on the HP IC-CAP circuit and device modeling software. Instrument drivers were written for instruments such as the HP 8114A pulse generator, the HP 54720A digital

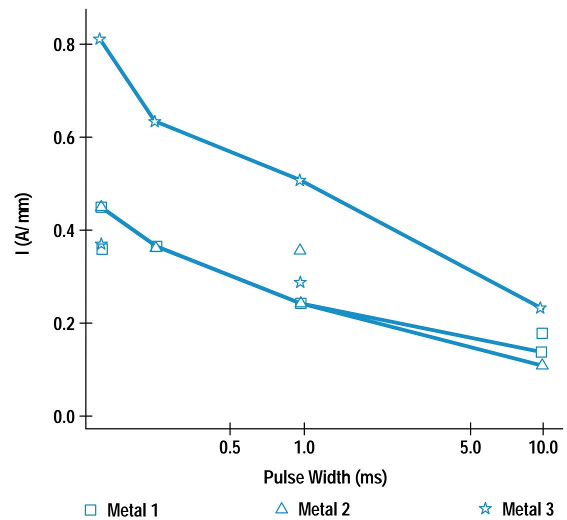


Fig. 10. Pulsed failure current per micrometer width in metal lines for an advanced three-level-metal CMOS process.

storage oscilloscope, and the Bertan 225 high-voltage power supply. The drivers and the IC-CAP environment were supplemented with additional C routines that, at the discretion of the user, enable the IC-CAP open measurement interface to indirectly control a switch matrix in the HP 75000 VXI mainframe system. The test system is currently used by the HP Integrated Circuit Business Division's Technology Development Center as a tool in the development of CMOS processes with built-in EOS/ESD robustness.

Acknowledgments

The author would like to acknowledge Ebrahim Khalily and Farshid Iravani from the Semiconductor Systems Group and Keith Moore from HP Laboratories for helpful discussions and comments. Thanks also to Rachel Wirth for providing the latest instrumentation available.

References

1. T. Green and W. Denson, "A review of EOS/ESD field failures in military equipment," *Proceedings of the EOS/ESD Symposium,* September 1988, pp. 7-13.
2. C. Diaz, S. Kang, C. Duvvury, and L. Wagner, "Electrical overstress power profiles: a guideline to qualify EOS hardness of semiconductor devices," *Journal of Electrostatics,* Vol. 31, 1993, pp. 131-144.
3. D. Merril and E. Issaq, "ESD design methodology," *Proceedings of the EOS/ESD Symposium,* 1993, pp. 233-237.
4. *MIL-STD-883C, Electrostatic Discharge Sensitivity Classification, Technical Report, Notice 8,* DOD, March 1989.
5. L. Roozendal, et al, "Standard ESD testing of integrated circuits," *Proceedings of the EOS/ESD Symposium,* September 1990, pp. 119-130.
6. R. Renninger, M. Jon, D. Ling, T. Diep, and T. Welsher, "A field-induced charge-device model simulator," *Proceedings of the EOS/ESD Symposium,* September 1989, pp. 59-71.
7. D. Pierce, W. Shiley, B. Mulcahy, K. Wagner, and M. Wunder, "An overview of electrical overstress effects on semiconductor devices," *Proceedings of the EOS/ESD Symposium,* September 1981, pp. 120-131.
8. A. Amerasekera, L. Roozendaal, J. Bruines, and F. Kuper, "Characterization and modeling of second breakdown in nMOSTs for the extraction of ESD-related process parameters," *IEEE Transactions on Electron Devices,* Vol. ED-38, September 1991, pp. 2161-2168.
9. T. Maloney and N. Khurana, "ESD on CHMOS devices—equivalent circuits, physical models and failure mechanisms," *Proceedings of the IEEE International Reliability Physics Symposium,* 1985, pp. 212-222.
10. C. Diaz, C. Duvvury, and S. Kang, "Circuit-level electothermal simulation of electrical overstress failures in advanced MOS I/O protection devices," *IEEE Transactions on Computer-Aided Design,* Vol. TCAD-13, 1994, pp. 482-493.