HEWLETT-PACKARD
JOURNAL

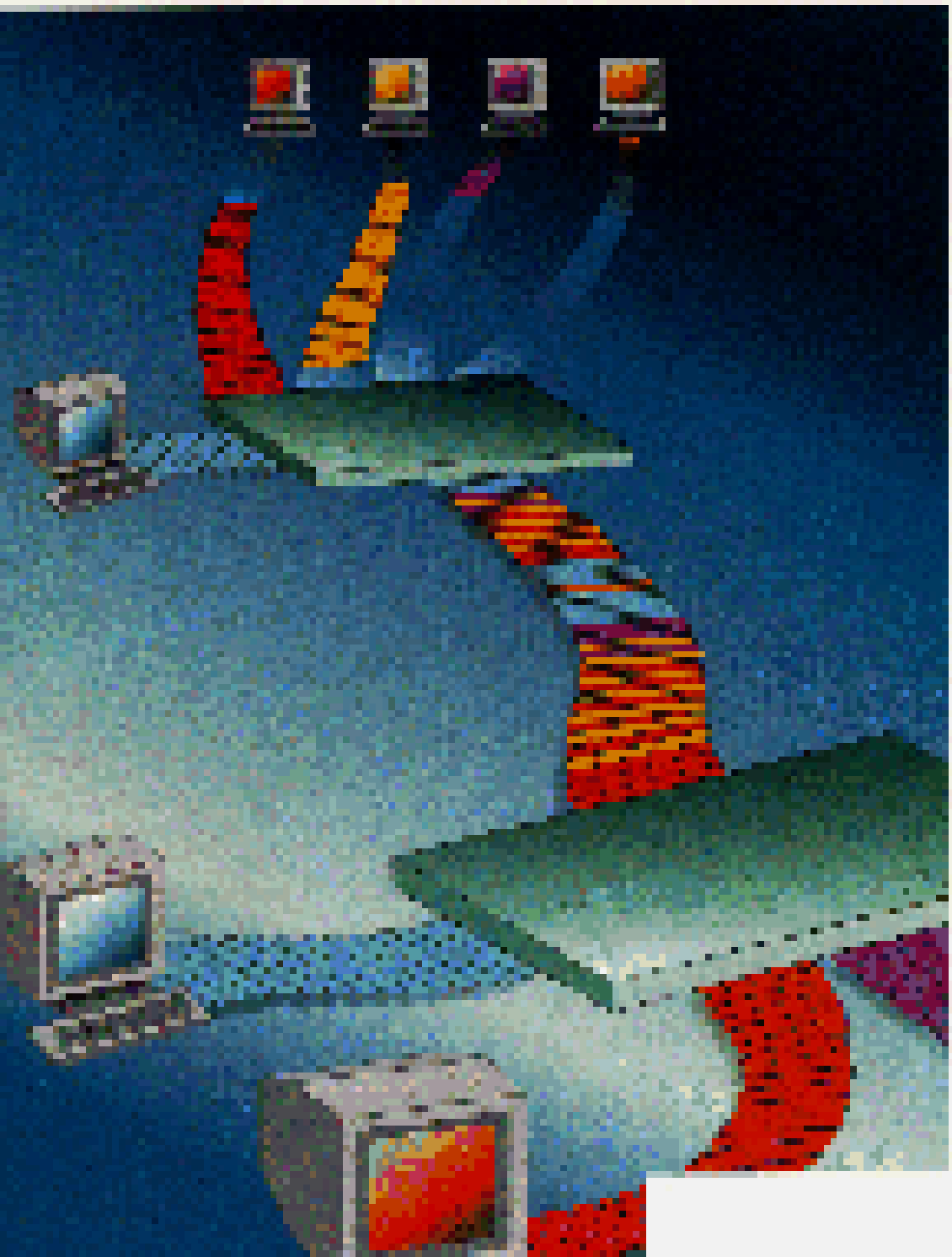


table of contents

August 1995,
Volume 46, Issue 4

Articles

Introduction to 100VG-AnyLAN and the IEEE 802.12 Local Area Network Standard

100VG-AnyLAN provides a 100-Mbit/s data rate with guaranteed bandwidth and maximum access delay for time-critical applications such as multimedia, using existing building wiring. It uses demand priority protocol. Developed by Hewlett-Packard and now supported by over 30 companies ranging from integrated circuit vendors to systems suppliers, demand priority is well on its way to becoming the IEEE 802.12 standard.

by Alan R. Albrecht and Patricia A. Thaler

100VG-AnyLAN is a new, high-speed addition to HP's AdvanceStack local area network (LAN) product group. It is an economically effective upgrade path for congested 10-Mbit/s 10Base-T Ethernet and 4/16-Mbit/s token ring networks. It provides a 100-Mbit/s data rate with guaranteed bandwidth and bounded access delay for time-critical applications, using existing building wiring. This provides high performance for traditional data transfer applications. It also provides emerging multimedia applications, such as interactive video, with the low delays they require. It delivers this performance over the most common networking medium, 4-pair unshielded twisted-pair (UTP) telephone wire.

100VG-AnyLAN uses the demand priority protocol, which was developed as a joint effort by Hewlett-Packard Laboratories in Bristol, England and the HP Roseville Networks Division in California. Now supported by over 30 companies ranging from integrated circuit vendors to systems suppliers, demand priority is well on its way to becoming the IEEE 802.12 standard.

The 100VG-AnyLAN articles in this issue look at the development of the demand priority protocol and the 100VG-AnyLAN product set.

Local Area Network Technology

Before the initial development of local area networking in the late 1970s, the telephone system was the only generally available data communications option. Bandwidth (3 kHz on a voice-grade line) was clearly a problem and a number of different types of new data networks were proposed. Two, Ethernet[1] and token ring,[2] have emerged to dominate the local area networking market.

Ethernet (IEEE 802.3). Ethernet was developed in the late 1970s as a 10-Mbit/s answer to the limitations of the telephone system. It became an IEEE standard in 1985. All nodes were connected to a single central coax bus, which proved to be somewhat inflexible as users change locations or are added to the network.

The Ethernet access policy is CSMA/CD, which stands for carrier sense, multiple access with collision detection. It allows any node to transmit a packet (with up to 1500 bytes of data) anytime it detects silence (no signal) on the network. This can lead to packet collisions if two or more nodes need to transmit and detect silence at the same time. Each involved node is required to back off (cease transmitting) immediately after a collision is detected, but time is consumed and the available bandwidth is effectively reduced during high-traffic periods.

The protocol also requires each node to monitor the network traffic and to decode (filter) the destination address of each packet to determine whether it should be received by the node. Packets with the node's individual or group address are copied into memory and packets with nonmatching addresses are ignored.

The 10Base-T star topology was proposed by Hewlett-Packard in 1987 and became part of the standard in 1990. The center of the star is a network concentrator (hub) which is typically located in a wiring closet. Each node is connected to the hub by voice-grade twisted-pair cable. 10Base-T retains the basic features and access policy of the bus network and also adds a level of fault tolerance. Link faults at individual nodes are isolated by the hub and do not take down the entire network. 10Base-T has become the most common IEEE 802.3 network configuration.

Token Ring (IEEE 802.5). Token ring was proposed as a 4/16-Mbit/s solution to the Ethernet collision problem and became an IEEE standard in late 1984. The original network structure is a ring around which both tokens and information packets (up to 4500 data bytes) are passed. The network medium is IBM type 1 shielded twisted-pair (STP) cable. Token ring networks are also now commonly installed in star configurations.

The token ring access policy is designed to be both collision-free and priority-based. It prevents any node that does not currently "own" the token from transmitting a data packet,

and it provides eight priority levels to allow some classes of data to take precedence over other classes.

All data packets and tokens contain an access control field that allows the successive nodes on the network both to reserve the token and to indicate their reservation priority level. The node that currently owns the token transmits its data packet with the reservation bits in the access control field set to minimum priority. Each successive node forwards the packet as it is being received. It also interrogates the destination address field to determine whether it should copy the data frame and the access control field to determine the current reservation level. If the node needs to send a data packet and the reserved priority level is less than the node's level, the node indicates its need by changing the value of the reservation bits in the forwarded packet.

The sending node removes the packet from the network and transmits a new token with the priority bits of the access control field set to the priority level indicated in the returned packet. The token then circulates to the node that first reserved that priority. That node removes the token and transmits a data packet. The token circulates continuously at minimum priority in an idle network.

The Local Area Networking Market

International Data Corporation (IDC), a market research firm, reports that the worldwide installed Ethernet base at the end of 1993 was 26,376,000 nodes, up 102% from 1992.[3] They predict an installed base of over 75,000,000 by the end of 1995, predominantly 10Base-T.

The token ring worldwide installed base was 6,744,000 at the end of 1993, up 115% from 1992.[3] IDC predicts that the token ring installed base will approach 14,900,000 by the end of 1995.

Current Network Pressures

The last ten years has seen a hundredfold increase in the speed of computers and the size of files created by sophisticated applications. Meanwhile, the data transfer rate of most networks has remained constant at 10 Mbits/s to 16 Mbits/s.

The first signs of network strain are users complaining that performance is falling off and response times are rising. The cause is almost always congestion—too many users, too much information. Network-connected high-performance desktop systems are intended to give users instant access to any appropriate information, anywhere in the organization, at any time, and this generates high levels of traffic.

Sometimes, network bottlenecks arise from individual applications exceeding the bandwidth of the network. Data-intensive applications, such as database access, image analysis, desktop publishing, network printing, and CAD, require that very large amounts of information be transferred in a single burst. For example, a desktop publishing application might require 10 megabytes for a single page incorporating several typefaces, a bitmapped logo, and four-color graphics. On a typical Ethernet or token ring network, it could take as long as 20 seconds to retrieve that one page. A multipage document could take a minute or two to retrieve.

Cable Types

Cables can be categorized in various ways: according to their physical structure, the material used for transmitting signals, and the uses for which they are suitable. Common types are listed below.

UTP: Unshielded twisted-pair, 100-ohm balanced cable. The lack of shielding makes UTP cable very low-cost, but introduces problems of cross talk when the pairs are in close proximity.

Category 3: Voice-grade cable, such as telephone wire, with 16-MHz bandwidth, used in 4-pair groups for each link. 25-pair bundles of Category 3 UTP are common in existing LANs. This is an important consideration when designing a network protocol.

Category 4: 20-MHz bandwidth, used in 4-pair groups for each link.

Category 5: Data-grade cable with 100-MHz bandwidth, used in 2-pair or 4-pair groups for each link.

Optical-Fiber: Cable consisting of a minimum of two strands of optical fiber running parallel within a protective jacket. Each fiber is usually composed of glass 125 μm in diameter, and has a 62.5-μm core. Transmission is by light beam at 850-nm or 1330-nm wavelength.

STP: 150-ohm balanced shielded twisted-pair cable. Usually used in 2-pair groups for each link.

The situation will rapidly worsen with the accelerated development of time-sensitive multimedia applications. Real-time audio and video for video conferencing and interactive video require that packets of data be transferred continuously with minimal delay. They cannot afford to have any packet delayed or dropped because of a collision or congestion on the network.

Design Goals

The problem presented to HP's network design team contained several major goals and considerations:

- Speed. The current networks are clearly too slow. A major improvement would be to increase the speed of the network. 100 Mbits/s would allow ten times the amount of traffic.
- Guaranteed Access. While multimedia and other time-sensitive applications require the increased total bandwidth that 100 Mbits/s would provide, they also need guarantees that information will get through within a stated delay window, whatever the traffic on the network.
- Cost. Existing networks have already required major investments in the wiring structure. The new network should be able to operate over generic twisted-pair building wiring. Fiber-optic cabling should also be allowed.
- Topology. To be compatible with existing wiring, the new network must use a star topology. The allowed network diameter should be at least 2.5 km with three or more levels of hub cascading.
- Software Compatibility. The network should be compatible with both Ethernet and token ring frame formats and should preserve existing investments in network and applications software.

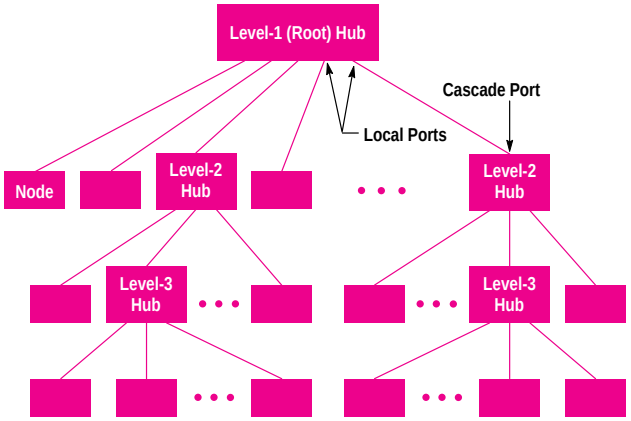


Fig. 1. A cascaded star network.

- Error Susceptibility and Detection. The implementation should have a physical layer bit error rate of less than 10^{-8} and a coding scheme that will guarantee detection of errors in any three bits within the data frame without compromising the cyclic redundancy check (CRC) defined for the IEEE 802.3 and 802.5 frame formats.
- Privacy. Both Ethernet and token ring provide address filtering at the node level, effectively allowing any node to copy any data packet sent on the network. To enhance the privacy of data communications, address filtering of individually addressed packets should be provided as an option within the hub.
- Robust Operation. Continuous operation of the network is required. Physical connections should be tested before allowing nodes to enter the network and provision should be made to allow the identification and removal of disruptive nodes.
- Network Management. An optional network management capability should be provided to monitor network performance, isolate faults, and control network configuration.

Demand Priority and 100VG-AnyLAN

The protocol for the new network is demand priority. It combines the best characteristics of both Ethernet (simple, fast access) and token ring (strong control, collision avoidance, and deterministic delay).

The HP product is 100VG-AnyLAN. It is based on chip technology developed by HP and AT&T that delivers an effective data rate of 100 Mbits/s over several different link configurations: 4-pair Category 3 unshielded twisted-pair (UTP) cable, 2-pair shielded twisted-pair (STP) cable, or 2-pair fiber-optic cable. UTP, STP, and fiber-optic cable can be intermixed on the same network. 100VG-AnyLAN can also operate with the 25-pair bundled cable that is used in many 10Base-T networks, as long as hubs are not connected through bundled cable.

Network Topology

The basic topology used by 100VG-AnyLAN networks is the star configuration. Each hub has two or more local ports and can optionally have one cascade port for connection to a higher-level hub. Nodes can be user stations, bridges to other networks, LAN analyzers, or lower-level hubs.

A network can contain several levels of hubs interconnected in a cascade as shown in Fig. 1. The topmost hub is designated as the level 1 (root) hub. Hubs in lower levels of the cascade are designated by the number of links between them and the root hub. The level number of any particular hub can be determined by the equation:

$$\text{Hub Level} = 1 + (\text{number of link segments away from the root hub})$$

Hubs at the same cascade level have the same level number.

The maximum number of nodes that can be connected to the network is dependent on the level and frequency of traffic each node generates. The maximum topology diameters and the number of levels of cascading are limited by the allowable delay between the node and the root hub, and can be calculated for any proposed configuration. When there is only one intermediate hub between the root hub and the node, for example, the maximum distance between a node and the root hub is 4 km. Each additional intermediate hub reduces the hub-to-node distance by 1.0 km, resulting in a maximum of four intermediate hubs and a root-hub-to-node distance of 1 km.

Local hub-to-node distances depend on the type of media used for the link: 4-pair UTP links configured with Category 3 or Category 4 cable and links configured with STP cable should not exceed 100 m. Category-5 links can be up to 150 m long. Fiber-optic links can be even longer: 500 m with 850-nm transceivers and 2000 m with 1300-nm transceivers.

Demand Priority Protocol

Control of a demand priority network is centered in the hubs and is based on a request/grant handshake between the hubs and their connected nodes. Access to the network is granted by the hub to requesting nodes in a cyclic round-robin sequence, based on the priority of the request. Within a priority level, selection of the next node to transmit is determined by its sequential location in the network rather than the time of its request. Data is encoded before transmission and is checked for errors at each intermediate hub and the receiving node. Either IEEE 802.3 or IEEE 802.5 frame format can be used.

Architectural Model

The demand priority protocol contains four sublayers corresponding to the two lower layers of the ISO Open Systems Interconnection (OSI) reference model shown in Fig. 2.

The functions of the OSI data link layer are implemented in two sublayers: the LLC and demand priority MAC sublayers. The upper sublayer in a network node is the IEEE 802.2 logical link control (LLC) sublayer. The media access control (MAC) sublayer provides data formatting and control of packet transmission (or reception) in the transmitting (or receiving) node. The MAC also initiates outgoing control requests and acts on received control indications.

Each hub provides control of its connected star portion of the network. The RMAC sublayer provides a superset of the functions of the node's MAC sublayer (except frame formatting). It selects which node will next be granted permission

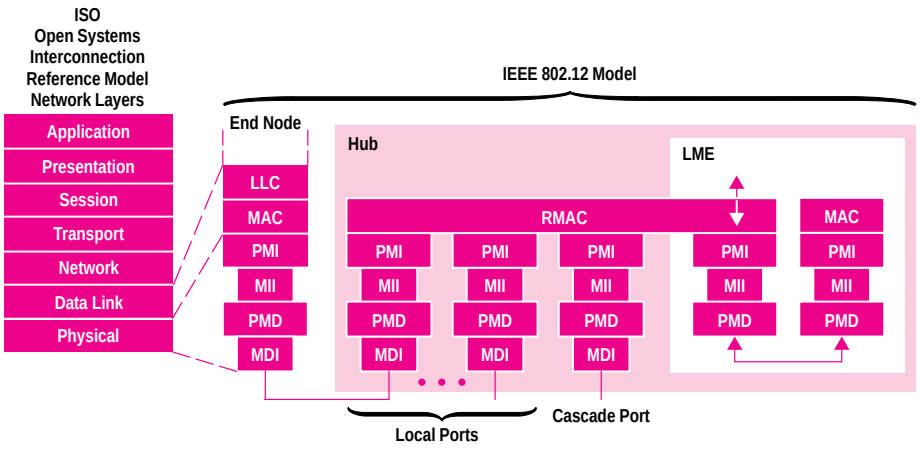


Fig. 2. The demand priority architectural model. LME is the layer management entity.

to send a packet, determines where the received packet will be sent, provides local control of packet reception and retransmission, and monitors each connected link for proper operation. The RMAC initiates outgoing control requests and acts on received control indications. There is no LLC sublayer in a hub.

The functions of the physical layer of the OSI model are also provided by two sublayers: the PMI and PMD sublayers. The physical medium independent (PMI) sublayer presents the same logical interface to both the MAC in a node and the RMAC in a hub. It provides the functions that are common to all link media: ciphering and encoding data and inserting stream headers and trailers before transmission, and removing stream headers and trailers and decoding and deciphering data during reception.

The physical medium dependent (PMD) sublayer contains the functions that are dependent on the particular link medium: bit encoding, signal conditioning, and multiplexing (if necessary) before transmission, and signal recovery, demultiplexing (if necessary), and bit decoding during reception. The PMD translates control requests and generates outgoing control signals. It also detects control signal transitions and generates the appropriate control indication for the MAC or RMAC.

The medium independent interface (MII) is defined in the draft standard as an optional, physically exposed connection to allow link configuration interchangeability (for example, changing from 4-pair UTP to a fiber link). The medium dependent interface (MDI) is the connector between the PMD and the link media.

Standards Development

The demand priority protocol is currently in the process of becoming the IEEE 802.12 standard. The following milestones had been accomplished by May 1995:

- November 1992. A proposal was made for a demand priority development project.
- July 1993. IEEE established the 802.12 Demand Priority Working Group.
- November 1993. An initial draft document (D1) was submitted to the 802.12 Working Group for review.
- July 1994. The IEEE 802.12 draft standard (D4) was submitted to the Working Group for ballot.
- January 1995. The IEEE 801.12 draft standard (D7) was submitted to the LAN/MAN Standards Committee for sponsor ballot.
- March 1995. Sponsor ballot was successfully completed.
- May 1995. The IEEE 802.12 draft standard (D8) was submitted to the Review Committee of the IEEE Standards Board. It is anticipated that the Review Committee and the Standards Board will vote on approving IEEE 802.12 in June.

As the base IEEE l802.12 standard approaches completion, the 802.12 Working Group is investigating future enhancements. At the March 1995 meeting, study groups were formed to investigate higher-speed operation (0.4 to 4 gigabits per second) and a PMD for 2-pair Category 5 UTP. Also under discussion are redundancy and full-duplex links.

100VG-AnyLAN Products

HP currently offers a variety of products for 100VG-AnyLAN within the HP AdvanceStack networking family:

- HP AdvanceStack 100VG Hub15 (HP J2410A) is a 15-port 100VG-AnyLAN hub (Fig. 3).
- HP AdvanceStack 100VG SNMP/bridge module (HP J2414A), when installed in the expansion slot of 100VG Hub15, adds SNMP network management and bridging to 10-Mbit/s Ethernet networks.

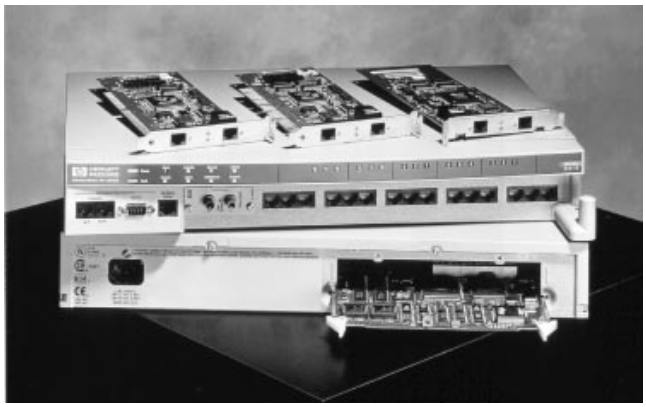


Fig. 3. The HP AdvanceStack 100VG Hub15 (HP J2410A) hub is a 15-port 100VG-AnyLAN hub.

Other Network Technologies

FDDI. Fiber Distributed Data Interface (FDDI) networks run at a standard 100-Mbit/s transfer rate using multimode optical-fiber cabling. The purchase cost is high, partly because it necessitates new cabling for existing networks. FDDI is also available as a high-speed backbone network connecting LANs.

A twisted-pair version, TP-PMD (twisted-pair physical medium dependent) FDDI, is under development, using Category 5 UTP and STP copper cables. This version is sometimes called Copper Distributed Data Interface (CDDI).

ATM. Asynchronous transfer mode (ATM) is a new network technology particularly suitable for wide area networks and campus backbones. It is intended to allow seamless integration of campus LAN backbones into the wide area network.

ATM uses cell switching (53 bytes per cell) similar to high-speed telephone switching over existing UTP or optical-fiber cabling. It runs at 25 to 622 Mbits/s.

Table I
Cabling and Topological Comparisons

	FDDI	ATM	100Base-T	Demand Priority
100-Mbit/s Category 3 cable supported?	No	50 Mbits/s with complex coding	Yes	Yes
Bundled cables supported?	No	No	No	Yes
Multiple cascades supported without bridging or routing?	Yes	Yes	No	Yes
Cost	High	High	Medium	Low

Ethernet and Token Ring Switching. Switching is also being introduced into Ethernet and token ring networks. To increase overall throughput, LANs are segmented. Hubs switch packets dynamically between connected segments allowing simultaneous transmissions among pairs of network segments. This increases bandwidth by two or more times that of individual segments.

100Base-T. 100Base-T is a scaling of CSMA/CD to 100 Mbits/s. There is no migration path or accommodation for existing token ring users. The technique cannot emulate 10Base-T topologies since the maximum topology is two repeaters.

Table II
System Comparisons

	FDDI	ATM	100Base-T	Demand Priority
Supports multimedia with guaranteed delay and bandwidth?	Yes	Yes	No	Yes
End-node adapter card complexity	Node management is expensive	Segmentation or reassembly of frames is expensive	Low	Low
Ethernet 802.3 networks can be upgraded without software changes?	No	No	Yes	Yes
Token ring 802.5 networks can be upgraded without software changes?	No	No	No	Yes

- HP 10/100VG selectable ISA, EISA, and PCI adapters (HP J2573A, J2577A, and J2585A) are PC LAN adapter cards with one RJ-45 connector for 10-Mbit/s 10Base-T and another for 100-Mbit/s 100VG-AnyLAN. The adapters automatically sense which network they are connected to and select the correct mode of operation.
- HP 100VG-AnyLAN/9000 (HP J2645AA, J2655AA) are adapters for HP 9000 Series 700 workstations. They support connection to either 10-Mbit/s 10Base-T or 100-Mbit/s 100VG-AnyLAN. The adapters automatically sense which network they are connected to and select the correct mode of operation.
- HP 100VG-AnyLAN development system (HP E2463A) is a development system for designers and operators of 100VG-AnyLAN network products. It finds the root cause of any IEEE 802.12 design and interoperability problems.

Upgrading Existing Networks

The current 100VG-AnyLAN product set provides a smooth, step-by-step way for customers who wish to upgrade their existing 10Base-T Ethernet networks. The only required elements are new hubs for the network and new adapter cards for each node. Existing network management interfaces, operating systems, bridges, and routers can remain.

In most cases, not all of the network will have to be upgraded at the same time. Consider, for example, an existing 10Base-T network with a mix of high-traffic and low-traffic users that all need to access file servers and printers. The first phase is to identify the high-traffic users to determine how many nodes need to be upgraded. This will determine the number of 100VG-AnyLAN hub ports and network adapters that will be required (including network file servers and printers).

The second phase is to acquire the necessary 100VG-AnyLAN hubs and network adapter cards (one hub for each 15 nodes if HP AdvanceStack 100VG-AnyLAN hubs are used), and to install them in the network as depicted in Fig. 4. The link cable is disconnected from each node that has been identified for upgrading, a replacement PC LAN adapter card is installed, and the link cable is reconnected.

100VG-AnyLAN hubs are installed adjacent to the existing 10Base-T hubs that service high-traffic users. The network cables to each upgraded node are disconnected from the 10Base-T hub and connected to a 100VG-AnyLAN hub (some rearrangement of the 10Base-T node-to-hub connections

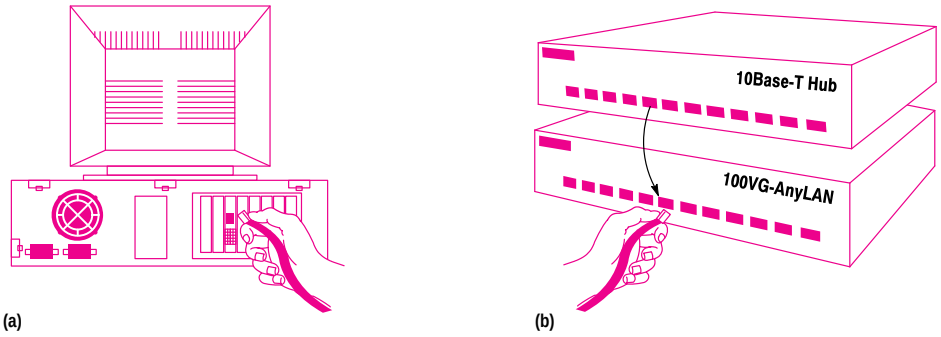


Fig. 4. Upgrading a 10Base-T network to 100VG-AnyLAN. (a) Upgrade adapter at the desktop. (b) Upgrade hub in the wiring closet.

may be advisable). A bridge module between the 100VG-AnyLAN and 10Base-T root hubs interconnects the two LANs.

The network is again ready for use with an upgraded topology as shown in Fig. 5. Additional 100VG-AnyLAN hubs and adapters can be added as needed to accommodate changing traffic levels in the remaining 10Base-T nodes.

The Future

For 100VG-AnyLAN to become a pervasive network topology, the concept of the protocol must be adopted by three major groups: network product manufacturers, network system designers, and network users. Customers must be able to obtain complete sets of the products that are required to meet their individual needs, and the products they obtain should be compatible with each other even if they are obtained from different companies.

An early measure of 100VG-AnyLAN's acceptance can be seen by the number of organizations involved in the demand priority standards development, and in the number adopting the protocol. By January 1995, the following companies had already begun delivering or had announced intention to deliver 100VG-AnyLAN products:

- IC chipsets. AT&T, Motorola, Texas Instruments, Applied Micro Circuits, and Pericom Semiconductor.
- Hubs. HP, Thomas-Conrad, Chipcom, NEC, MultiMedia LANs, Compex, Alfa, Katron Technologies, Optical Data Systems, Anritsu, D-Link, and Ragula Systems.
- Node Adapters. HP, Thomas-Conrad, NEC, Compex, Alfa, Katron Technologies, Ragula Systems, Anritsu, D-Link, Interphase, Optical Data Systems, and Racore Computer Products.
- Multimedia Networking Software. Starlight Networks.
- Development Systems. HP and AT&T.
- Switches and other Internetworking Products. Cisco, Compex, Newbridge Networks, Optical Data Systems, and Plain Tree Systems.
- Desktop Systems. IBM, HP, Compaq, and NEC.

The current 100VG-AnyLAN products provide customers with ten times the speed and up to 16 times the throughput of a 10Base-T Ethernet network at about twice the price. Development of VLSI devices that provide greater integration of the physical layer and the MAC should lead to even lower-cost products.

Demand priority has been designed to be architecturally independent of any particular implementation technology. As such, future generations of demand priority networks may provide higher data rates.

Other Articles in this Issue

The following 100VG-AnyLAN articles provide more detail for their respective areas:

- Demand Priority. The article on page 13 introduces the round-robin pointers, priorities, and bandwidth allocation capabilities of the protocol. A typical demand priority transmission is described in step-by-step fashion, and results of performance simulations are provided.
- Physical Signaling. The article on page 18 gives an expanded description of the physical sublayer. Several design decisions leading to the development of demand priority as a replacement for 10Base-T are described. Quartet signaling, cross talk avoidance, and control signal generation and detection are explained. Differences between the 4-pair UTP and the STP and fiber-optic PMD and link requirements are discussed.

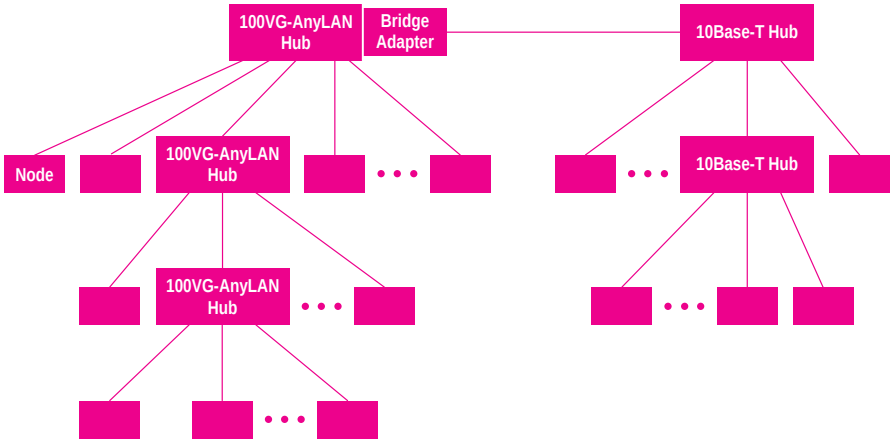


Fig. 5. Hybrid network consisting of interconnected 100VG-AnyLAN and 10Base-T networks.

- Coding. The article on page 27 looks at the techniques of coding data to provide error detection and electrical balance while optimizing the efficiency of the network. Additional insight into the reasons behind the decisions to include both data ciphering and encoding are given along with an expanded discussion of the properties of the coding scheme.
- Multimedia Requirements. The article on page 33 examines the specific demands multimedia applications make on any network system and shows how 100VG-AnyLAN addresses them.

Acknowledgments

The development of 100VG-AnyLAN from public proposal to draft standard and product release has taken only two years. This is an incredibly short time, and has involved almost Herculean efforts of the Roseville and Bristol teams. 100VG-AnyLAN was able to progress from investigation to product in such a short time because of the efforts of a fantastic team. We would particularly like to acknowledge the contributions of our fellow inventors: Alistair Coles, Simon Crouch, David Cunningham, Joe Curcio, Dan Dove, Steve Goody, Jonathan Jedwab, Michael Spratt, and Greg Watson. Although the two halves of the team live 5000 miles apart, these engineers, mathematicians, and physicists collaborated closely to produce a high-performance network.

Bill Lane, a professor emeritus from the California State University at Chico, contributed his skills as an editor and writer to produce the text for the draft standard and assisted in the editing of several articles in this issue of the HP Journal.

Completing a major development in a short time requires more than individual commitment. Thus, we also thank our management on both sides of the Atlantic—David Dack, Mark Gasteen, Robert Gudz, Dave Harris, Bill Kind, Gary McAnally, Tim McShane, Carolyn Ticknor, and Steve Wright—for their trust and wholehearted support. They always ensured that we had what we needed to succeed.

References

1. *Information Technology—Local and Metropolitan Area Networks—Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Specifications*, ISO/IEC 8802-3:1993.

2. *Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks—LAN/MAN Specific Requirements—Part 5: Token Ring Access Method and Physical Layer Specifications*, ISO/IEC DIS 8802-5:199?.

3. *Local Area Networks—Analysis of the Intelligent Hub Market*, International Data Corporation, May 1994.

Demand Priority Protocol

In multiple-hub networks, demand priority ensures fairness of access for all nodes and guarantees access time for multimedia applications.

by Alan R. Albrecht, Michael P. Spratt, Patricia A. Thaler, and Gregory C. A. Watson

Demand priority is the media access control protocol defined in the IEEE 802.12 draft standard. It is called 100VG-AnyLAN by HP. Various techniques are used to ensure fairness of access for all nodes and to guarantee access time for individual nodes. Round-robin selection procedures are used to give each node an equal opportunity to transmit data. Two priority levels are provided so that time-critical traffic such as interactive video, audio, and multimedia can be given priority service with guaranteed low delay. A bandwidth allocator can be introduced to control the amount of bandwidth each application can use.

These techniques make demand priority able to accommodate multimedia applications with guaranteed performance levels, while allowing normal traffic to use the remaining bandwidth in a fair manner.

Hubs

100VG-AnyLAN networks are centered around the concept of intelligent hubs (called repeaters in the IEEE 802.12 draft standard), each of which is at the center of a star configuration. A hub has several local ports, which are connected to individual nodes, which can be workstations, servers, bridges, routers, or lower-level hubs. A hub may also have one cascade port for connection to a higher-level hub.

Network Control

Control in a demand priority network is based on a request/grant handshake between the node and the hub. Each node needing to send a frame indicates its need to transmit by sending a request signal to the hub and waiting for the hub to grant it permission to transmit the frame. The concept can be seen as a sophisticated and flexible successor to the token in token ring networks; it ensures that collisions cannot occur in the network.

Hubs use a round-robin node selection procedure to ensure a fair opportunity for nodes to transmit data. The purpose is to ensure that no node can send two successive normal-priority frames until all other nodes have also had an opportunity to send a frame at normal priority.

Round-Robin Node Selection

The nodes are numbered in order around the entire network. The hub maintains a round-robin pointer that indicates which node is next to receive an opportunity to transmit. For example, if nodes 2, 3, 5, and 9 have indicated that they have packets to send and the round-robin pointer currently points to 3, node 3 gets to transmit. The pointer then moves on to 4. Assuming node 4 still has no data to send, node 5 will be selected next. The pointer will then move on to 6. When the pointer reaches the last node, it cycles back to 1.

The normal-priority selection cycle will be temporarily suspended when any node lodges a high-priority request. A separate round-robin pointer is maintained for high-priority requests, so that they too are treated fairly. The decision as to which node is next is made immediately after a transmission has finished. The hub looks at all requests that have been received since it last made a selection. Earlier requests are not stored so that spurious requests caused by noise or previous requests from a node that has changed its mind do not cause a false selection.

Typical Transaction

A typical transaction within a network using the demand priority protocol could run as follows. The example is for a network with a single hub, using 4-pair UTP cables. 4-pair UTP links require all four pairs to be used for data transmission in either direction, but dedicate two pairs in each direction for control when data is not being sent. Fig. 1 illustrates each step in the transaction.

While there is no data to be transmitted, all nodes send IDLE to the hub, and the hub returns IDLE to each node.

Suppose that node 1 has a frame to transmit to node 3 alone. It sends a request (REQ_N for normal priority, REQ_H for high priority) to the hub.

After referring to its round-robin pointer and checking for any other request, the hub decides to accept transmission from node 1. It indicates the selection by ceasing its IDLE transmission to node 1 (indicating a grant). This clears the outgoing signal from the hub and readies the link to receive data on all four pairs. Simultaneously, the hub also changes the control signal being sent to all other nodes from IDLE to INCOMING. (At this stage, the hub has no idea of the destination of the data packet.)

The other nodes respond to INCOMING by ceasing transmission to the hub. This clears their outgoing control signal so that they are prepared to receive the packet. The source node (node 1) starts transmitting its packet to the hub on all four pairs of wires as soon as it detects that the IDLE control signal has gone silent.

The hub decodes the beginning of the packet to determine the destination node or nodes. It finds that, in this case, the packet is unicast (destined for one node only) for node 3. It immediately starts sending the packet to node 3 and changes the control signal to all other nodes to IDLE.

Transmission continues until it is complete. Meanwhile, nodes not involved in the transfer can send REQ if they have packets to send. The hub notes such requests, but does not yet select the next node to transmit. When the source node (node 1) completes its transmission, it sends IDLE, or another REQ if it has another packet to send. After the hub has finished forwarding the frame, it selects the next node to transmit according to the round-robin selection procedure.

There is a possibility that a node that has received several consecutive packets and needs to make a transmission of its own might miss its turn in the round-robin cycle. This is because at the instant when the hub selects the next node to transmit, the receiving node is still receiving the end of the last packet. To overcome this, receiving nodes needing to transmit ignore the INCOMING control signal for a short period (called the *request window*) during which they submit their own request. The request window is shorter than the minimum gap between successive packets.

If the packet has a multicast address (it is destined for more than one node), the hub may need to wait until it has received the entire packet before transmitting it to the destination nodes. This is to avoid cross talk problems in 25-pair bundled cable. The article on page 18 provides more detail on 4-pair UTP links and the use of bundled cable.

In a network using 2-pair STP or fiber-optic cables, packets are transmitted down one fiber or pair of wires by multiplexing the four data streams. This means that the other fiber or wire pair is always available for lodging requests to the hub. See the articles on pages 18 and 27 for an expanded discussion of STP and fiber links.

Networks with Cascaded Hubs

A network can have several hubs in a tree structure known as a cascade (see Fig. 1 on page 8). One hub is designated the *root hub*. All other hubs will each have a connection to a next higher-level hub through their cascade port. They may also have lower-level hubs connected to some of their local ports.

Where there is more than one hub in a network, at any moment exactly one of them has *control* of the network activity. A hub is said to have control if it is selecting the next node to transmit. When the entire network is idle, the root hub has control.

If a network consists of a cascade of hubs, the round-robin pointers effectively treat all end nodes as if they were connected to a single hub, so that all nodes have an equal opportunity to transmit, however far they may be from the root hub. Fig. 2 shows the order of node numbering in a cascaded network.

A different control signal is needed when a hub receives a high-priority request while another hub is servicing normal-priority transactions. Suppose hub A is servicing normal-priority requests when hub B receives a high-priority request (see Fig. 3). Hub B sends the high-priority request up the cascade of hubs until it reaches hub X, which earlier passed control to hub A. Hub X then sends an ENABLE_HIGH_ONLY signal to hub A.

The ENABLE_HIGH_ONLY signal tells hub A to suspend its round-robin sequence at the end of the current transmission and return control to hub X. At the same time hub A tells hub X whether it had finished its normal-priority round-robin cycle. Hub X then passes control to hub B to service the high-priority request. When that is complete, assuming that hub A had not finished its portion of the round-robin normal-priority selection cycle, and assuming that no other high-priority requests have been received by hub X, control is passed back to hub A.

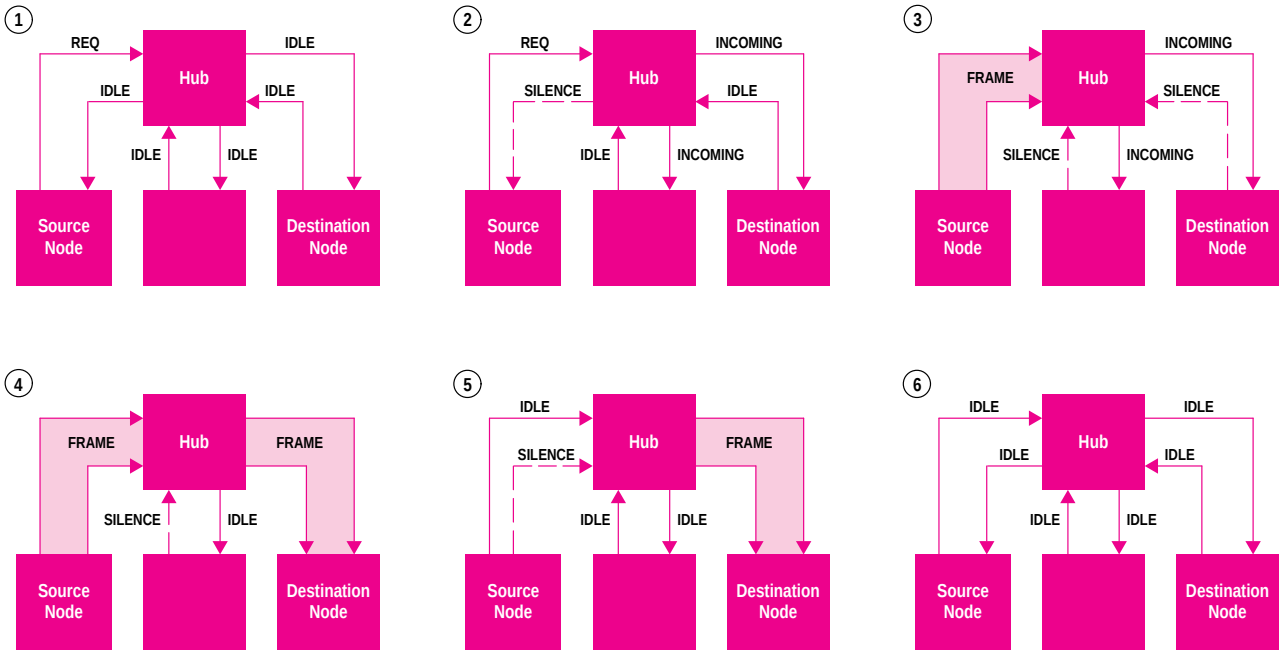


Fig. 1. The stages in a typical demand priority transaction.

Network Protocol Layers

The Open Systems Interconnection (OSI) Reference Model of the International Organization for Standardization (ISO) describes the structure of networks. It defines the seven layers shown at the left side of Fig. 2 on page 9. The lower three layers are relevant to the demand priority protocol.

Layer 3: Network Layer (NL). The network layer is responsible for passing a packet of data through an internetwork, which can consist of many individual LANs and even wide area links.

Layer 2: Data Link Layer (DLL). The data link layer provides the transmission of data between two nodes on the same network. It receives a packet from Layer 3, the network layer, and adds source and destination addresses and other information to it. The DLL consists of two sub-layers: the logical link control (LLC) sublayer and the media access control (MAC) sublayer.

The LLC sublayer links the network layer to the MAC. There are standard LLCs, enabling different protocols to link successfully. In 100VG-AnyLAN, the LLC can be either an IEEE 802.2 Class I LLC, supporting Type 1 unacknowledged, connectionless-mode transmission, or a Class II LLC, supporting Type 2, connection-mode transmission.

When a frame is ready for transmission in an end node, it is sent from the LLC sublayer to the MAC sublayer where the appropriate Ethernet or token ring MAC frame is built. The frame is then passed to the physical layer's PMI (physical medium independent) sublayer.

When a data packet is received from the physical layer, the MAC sublayer reassembles the MAC frame and performs various checks for errors in the received frame. Only valid frames are sent on to the LLC sublayer in an end node.

Frames received by the hub's RMAC sublayer are forwarded to the addressed destination (if it can be determined) and to all promiscuous ports regardless of error condition (there is no LLC in a hub). Frames containing errors are marked with an invalid packet marker.

Layer 1: Physical Layer (PHY). The physical layer defines the procedures and protocols associated with the physical transmission of bits (such as cable interfaces, data signal encoding, and connector types and pinouts). It consists of two sublayers: the PMI sublayer and the physical medium dependent (PMD) sublayer.

The PMI sublayer includes provisions for quartet signaling, data ciphering, 5B/6B encoding, the addition of preambles, and start and end frame delimiters (SFD and EFD). See the articles on pages 18 and 27 for details.

During transmission, the PMI sublayer accepts data from the MAC sublayer and prepares the packet for transmission. It converts the octet data into quintets which are separated into four streams. In each stream, each quintet is ciphered and then encoded as a 5B/6B sextet. The PMI adds physical layer headers and trailers to each data stream.

When receiving data packets, the PMI removes physical layer headers and trailers and then passes the packet onto the MAC layer. It decodes each received 5B/6B sextet, deciphers the resulting quintet, merges the four deciphered quintet streams, and converts the result into a single octet stream for delivery to the MAC sublayer.

The PMI sublayer connects with the PMD sublayer through the medium independent interface (MII).

The PMD sublayer includes channel data packet multiplexing (for 2-pair STP and fiber-optic cabling only), NRZ encoding, link medium operation, and link status control. It connects with the physical medium (the cable) through the medium dependent interface (MDI).

Link Initialization—Joining the Network

When a node first joins the network, a handshake training sequence occurs between it and the hub, during which the node sends the hub its 48-bit MAC address. It also sends other information, such as what type of frame it will use (IEEE 802.3 or IEEE 802.5) and if it wants to receive all packets of data whether addressed to it or not (promiscuous mode). The hub accepts or rejects this information. For example, the hub might be configured by the network administrator to reject promiscuous mode to preserve a high level of data privacy.

Guaranteed Performance

The two priority levels make it possible to guarantee bandwidth to applications and to keep the access delay (the time for which nodes may have to wait before being allowed to transmit) within bounds.

Bandwidth and access delay depend on the size of the network and the way it is configured. Nodes using high-priority traffic may need to be configured so that the amount of bandwidth they can use is restricted, but nodes using only normal-priority traffic will operate totally unaware that a high-priority service is being provided to other nodes.

Bandwidth

Without any other form of control, nodes wishing to send high-priority traffic will automatically be allowed to send an equal number of high-priority frames. Any bandwidth not used by this high-priority traffic is then automatically shared

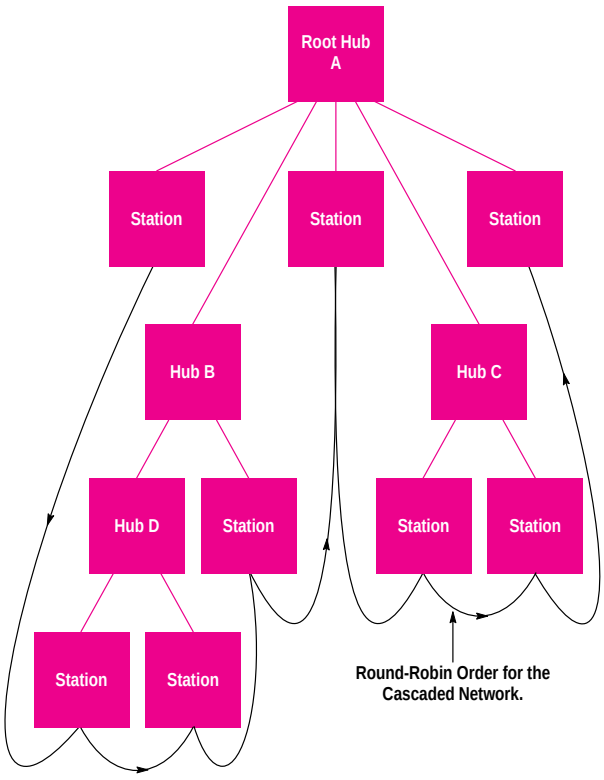


Fig. 2. The numbering of nodes in a cascade of hubs.

between all nodes wishing to send normal-priority traffic. Clearly, where there is excessive high-priority traffic, the bandwidth allowed for high-priority requests needs to be restricted in some way so that normal-priority traffic is never completely stopped. Ways of doing this are described in the article on page 33.

Access Delay

The deterministic behavior of the round-robin selection procedure means that it is easy to establish a reasonably accurate estimate of the maximum access delay that a high-priority packet will experience. For a hub with n nodes using high-priority traffic, the worst-case delay will be nT, where T is the time it takes to transmit the largest possible frame. It is not (n − 1)T as might be expected, because the worst case occurs when a high-priority request is received from all nodes just at the moment when the hub starts to service a normal-priority request. For example, the worst-case delay for a hub with 32 nodes forwarding 1500-byte IEEE 802.3 frames is 4 ms, or for 4500-byte IEEE 802.5 frames, 12 ms.

Access delay for a normal-priority request is more difficult to calculate, but the performance data presented in the next section shows what delays might be expected under heavy load.

Performance in a Simulated Example

Various simulations have been run to examine the effect on high-priority traffic as normal-priority traffic increases. Each simulation ran for 100,000 frames, and the results were taken from the central portion of the run to avoid anomalies that occur at startup and shutdown.

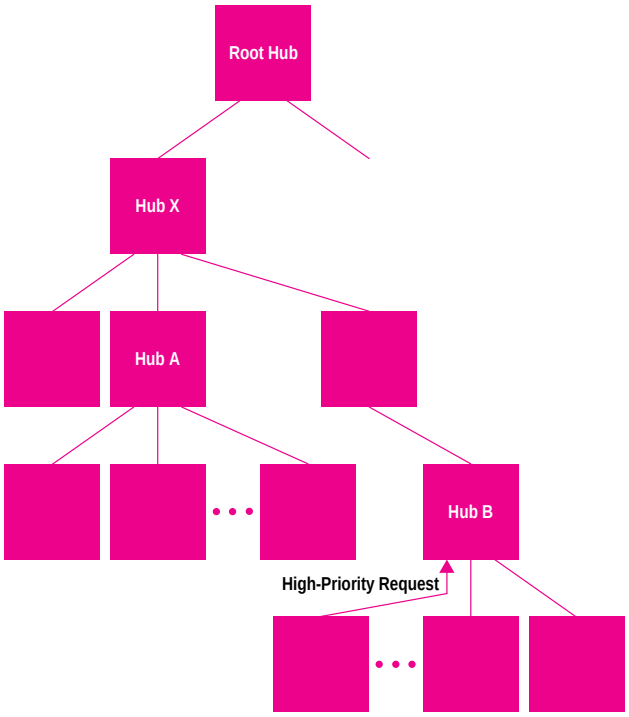


Fig. 3. In a cascade of hubs, a hub (hub B) may receive a high-priority request while another hub (hub A) is servicing normal-priority transactions. Hub B sends the high-priority request up the cascade of hubs until it reaches hub X, which earlier passed control to hub A. Hub X then sends an ENABLE_HIGH_ONLY signal to hub A.

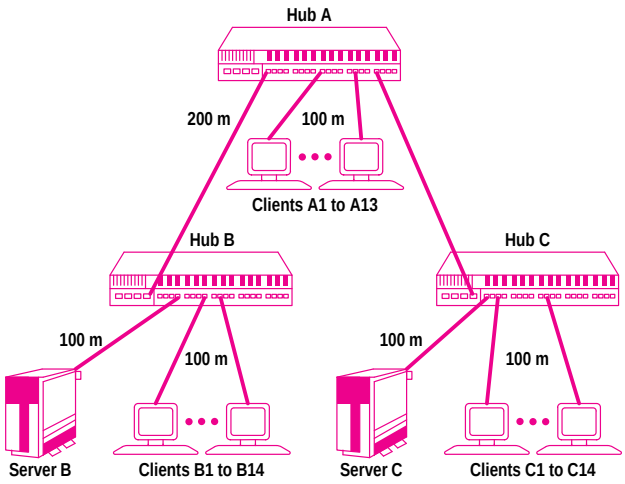


Fig. 4. Network configuration used in the simulated example.

The network configuration used for one of the simulations is shown in Fig. 4. All fifteen ports are used in each of the hubs. The hubs are connected to each other by 200-m links and the end nodes are connected to the hubs by 100-m links. Hubs B and C each have three nodes that are allocated high-priority bandwidth so that each node can send a block of eight maximum-sized frames every 10 ms. The bandwidth assigned to each node is equivalent to about 9.7 Mbits/s, and is substantially more than needed for MPEG-encoded video. The total high-priority traffic is therefore about 58 Mbits/s.

All stations, including the six with high-priority traffic, send some maximum-size frames at normal priority. The simulation measured the time for which nodes waited for access as the number of these normal-priority frames increased.

Fig. 5 shows the mean and maximum access delays that were observed for both high-priority and normal-priority traffic. Access delay is defined as the time a frame spends at the head of its node's transmission queue waiting for access to the network.

It is clear from the graph that the delay that high-priority traffic encountered was almost independent of the amount of normal-priority traffic. Even when the total load on the network was 93 Mbits/s (35 Mbits/s at normal priority in

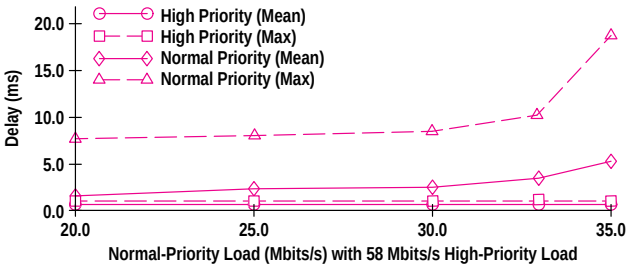


Fig. 5. Access delays in the simulation.

addition to the 58 Mbits/s of high-priority load), high-priority traffic still had a mean access delay of less than 0.5 ms and a maximum access delay of less than 0.8 ms.

These delays are, of course, dependent on the number of stations that have high-priority traffic, but it is clear that a number of high-priority streams can be supported with a guaranteed low delay.

Acknowledgments

We would particularly like to acknowledge the contributions of our fellow team members from Roseville Networks Division: Joe Curcio, Dan Dove, and Steve Goody.

Bibliography

Additional information on the demand priority protocol is contained in the following papers:

1. A. Albrecht, J. Curcio, D. Dove, S. Goody, and M. Spratt, "An Overview of IEEE 802.12 Demand Priority," *Proceedings of GLOBECOM*, 1994.

2. G. Watson, A. Albrecht, J. Curcio, D. Dove, S. Goody, J. Grinham, M. Spratt, and P. Thaler, "The Demand Priority MAC Protocol," *IEEE Networks Magazine*, January 1995.

Physical Signaling in 100VG-AnyLAN

A physical layer has been developed for demand priority local area networks that accommodates different cable types by means of different physical medium dependent (PMD) sublayers. The major goal was to provide 100-Mbit/s transmission on existing cables, including Category 3, 4, and 5 UTP, STP, and multimode optical fiber.

by Alistair N. Coles, David G. Cunningham, Joseph A. Curcio, Jr., Daniel J. Dove, and Steven G. Methley

The physical layer (PHY) of a 100VG-AnyLAN demand priority local area network (LAN) acts as an interface between the MAC (media access control) and the link (the cable), adding control signaling and data formatting to the MAC frame when necessary (see Fig. 1).

Several goals were identified in the early stages of the demand priority PHY development. First, the PHY should be as simple as possible, easy to implement, and above all, low in cost. Second, the PHY should provide robust data transfer. LAN performance deteriorates if multiple retransmissions of packets are necessary because of errors. Errors can occur if a PHY does not provide sufficient immunity against noise on the transmission medium (such as impulse noise on unshielded cable running close to switching gear). Typically a LAN is required to operate with less than one error in 10^8 bits.

Third, the PHY should support a range of existing media types. 10Base-T LANs operate over voice-grade, or Category 3, unshielded twisted-pair (UTP) wire. More recently, higher-quality UTP (Categories 4 and 5) has been specified and is now being used in new installations. Shielded twisted-pair (STP) has been used extensively in token ring LANs, although recently these too have been connected with UTP. Multimode optical fiber is also being used increasingly.

Fourth, the PHY should be capable of data transfer at 100 Mbits/s. Finally, when not transmitting data, the PHY should be capable of signaling five independent control states from one end of a link to another. These control signals are required for the operation of the demand priority protocol.

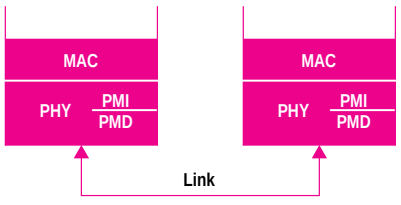


Fig. 1. Network layer model, showing the media access control (MAC) and the physical layer (PHY) with its physical medium dependent (PMD) and physical medium independent (PMI) sublayers.

The following constraints were also placed on the design of the demand priority PHY:

- EMC. The transmission techniques used for the PHY must not cause radiated emissions from the LAN equipment or cabling that would violate electromagnetic compatibility (EMC) regulations as shown in Fig. 2 (for example, FCC Class A in the U.S.A., EN 55022 in Europe). This constraint becomes most significant with Category 3 UTP.
- Cable cross talk. 25-pair bundles are often used for connections from a wiring closet to multiple wall outlets. As a result, several end nodes may be connected to a hub through a single 25-pair bundle. Simultaneous transmissions between the hub and more than one such end node may then cause cross talk within the cable (see "Cross Talk in Unshielded Twisted-Pair Cables" on page 19). The PHY must not be detrimentally affected by this cross talk.
- Transformer Coupling. Connections to twisted-pair (UTP and STP) are made through transformers so that dc currents cannot flow between devices with nonequipotential grounds. However, the transformers cause distortion of data signals with dc content, so the PHY must process the data to reduce the dc content of the signal. This often requires that some form of block coding be performed on the data before transmission.[1]

(continued on page 20)

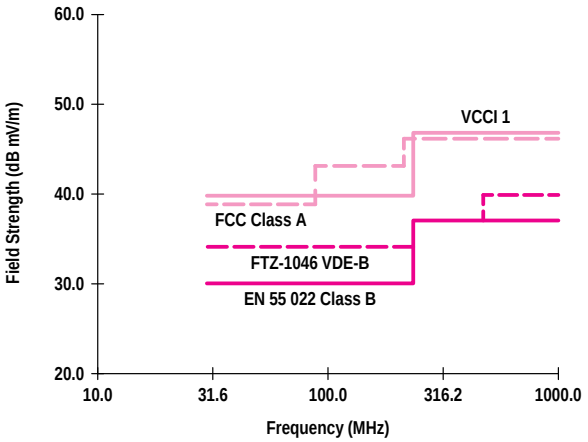


Fig. 2. CISPR and FCC radiated emissions regulations. Radiated emissions must fall below the lines shown for each regulatory body.

Cross Talk in Unshielded Twisted-Pair Cables

Cross talk in UTP cables is caused by capacitive coupling between pairs. Signals on pair A cause noise signals on pair B, and often the cross talk noise proves to be the limiting factor in the link performance. Cross talk occurs in two ways. Near-end cross talk (NEXT) happens when a signal from a transmitter at one end of a cable interferes with a receiver at the same end of the cable. Far-end cross talk (FEXT) occurs when a signal interferes with a receiver at the opposite end of the cable from the transmitter.

Near-End Cross Talk (NEXT)

Near-end cross talk loss is defined as:

$$\text{NEXT} = -20\log\left(V_n/V_i\right),$$

where V_n and V_i are shown in Fig. 1a. The minimum NEXT loss between pairs in a cable tends to follow a smooth curve, as shown in Fig. 1b, decreasing at a rate of 15 dB per decade. However, the actual NEXT between two particular pairs deviates significantly from this curve because of resonances in the twisted-pair. Typical measurements of the NEXT loss between some pairs in a 25-pair cable are also shown in Fig. 1b.

Far-End Cross Talk (FEXT)

Far-end cross talk loss is defined as:

$$\text{FEXT} = -20\log\left(V_f/V_i\right),$$

where V_f and V_i are shown in Fig. 2a. The minimum FEXT loss also decreases with frequency following a smooth curve, but at a rate of 20 dB per decade. As with NEXT loss, the actual FEXT loss between two particular pairs deviates from this curve. Typical measurements of the FEXT loss between some pairs in a 25-pair cable are shown in Fig. 2b.

Cross Talk Measurements

Our analysis of cross talk required a database of accurate and detailed measurements of cross talk between pairs in 25-pair cables. A measurement system was constructed to measure NEXT and FEXT losses of all pair combinations in 25-pair cables (see Fig. 3, next page).

Individual pairs were routed to the stimulus and response ports of a network analyzer via a computer-controlled switch. This allowed the automatic selection of 300 different pair combinations for NEXT measurements and 600 pair combinations for FEXT measurements. Any pair not being measured was terminated in 100 ohms via a balun and a 50-ohm termination internal to the switch. The network analyzer measured the cross talk loss (phase and magnitude) to 40 MHz, and this was downloaded to a computer database. Using this system, the NEXT and FEXT losses were measured for many thousands of pair combinations in a selection of 25-pair cables of varying manufacturer and age. The database was used to input NEXT and FEXT loss characteristics to the computation of cross talk noise described in "Cross Talk Analysis" on page 22.

(continued on next page)

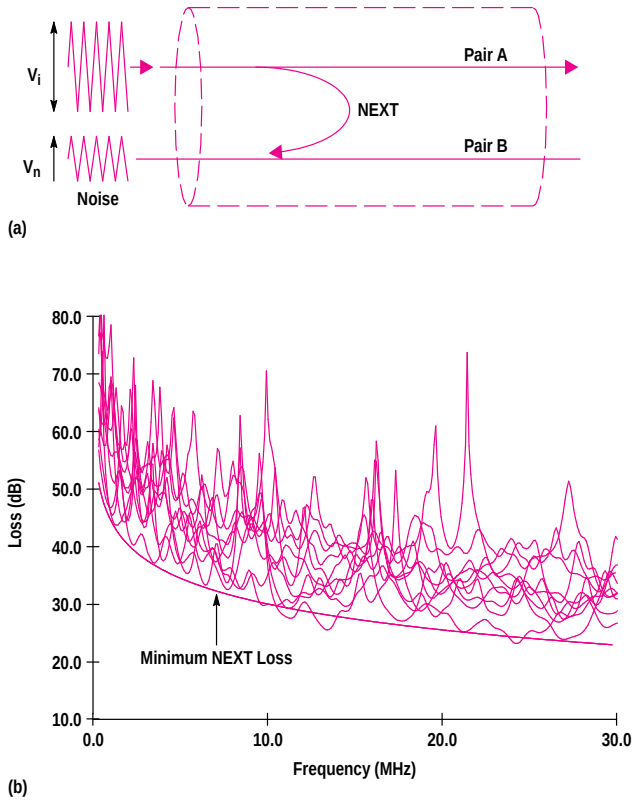


Fig. 1. (a) Near-end cross talk (NEXT). (b) Minimum theoretical NEXT loss and actual measurements.

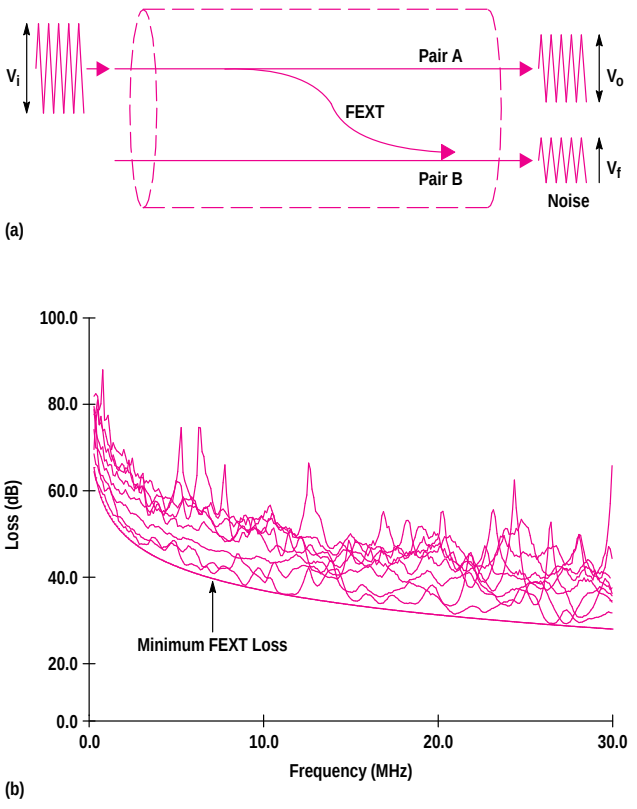


Fig. 2. (a) Far-end cross talk (FEXT). (b) Minimum theoretical FEXT loss and actual measurements.

(continued from page 19)

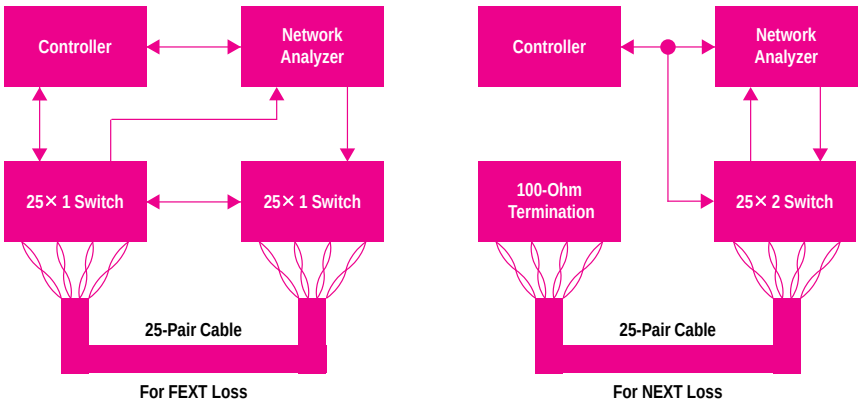


Fig. 3. Measurement system for NEXT and FEXT loss.

(continued from page 18)

Given these constraints, the design goals for the PHY were-met by developing a physical medium dependent (PMD) sublayer for each media type (UTP, STP, and multimode optical fiber) and a physical medium independent (PMI) sublayer that contains all the functions common to all media types. These two sublayers together form the demand priority PHY.

In the remainder of this article we will discuss the design choices made for each of the three demand priority PMDs.

The Four-Pair UTP PMD

The first PMD to be developed was to support UTP cabling, since this addresses the large 10Base-T upgrade market.

10Base-T uses full-duplex signaling at 10 Mbit/s on UTP cabling.[2] One twisted pair is used to transmit and one to receive data. The dc content of the signal is minimized by Manchester coding the data before transmission on the twisted-pair channel. Manchester coding is a very simple form of block coding:

Bit	Code
0	10
1	01

This is a 1B/2B block code, and results in a 100% bandwidth expansion. Manchester coding is spectrally inefficient relative to other block codes, but does provide a guaranteed transition for every two symbols, has very low dc content, and is very simple to implement.

The transmission rate in 10Base-T is 20 megabaud, giving a data rate of 10 Mbits/s (1 baud is one symbol per second). This means that the transmitted signal can be low-pass filtered with a cutoff frequency somewhat less than 20 MHz. This helps minimize radiated emissions above 30 MHz, the lower bound of stringent EMC regulations.

From 10Base-T to 100VG-AnyLAN

The progression from 10Base-T to 100VG-AnyLAN was made in three simple steps.

First, it was recognized that since full-duplex transmission was not absolutely necessary in a hub-based network, both twisted pairs used for 10Base-T could be used simultaneously for transmission in one direction or reception in the other. This immediately doubles the bit rate achieved on existing 10Base-T networks.

Second, the spectrally inefficient Manchester code was replaced with a more efficient 5B/6B block code (5 bits of data are coded to 6 transmitted binary symbols). This reduced the bandwidth overhead from 100% with Manchester coding to 20%. As a result, the data rate on each pair could be increased to 25 Mbits/s (that is, a symbol rate of 30 megabaud after 5B/6B coding) while the main lobe of the data spectrum remained below 30 MHz, the lower limit of EMC regulations. As with 10Base-T, low-pass filtering with a cutoff below 30 MHz can then be applied to minimize the risk of excessive radiated emissions. The 5B/6B code chosen also has very low dc content, which avoids distortion from the coupling transformers. See the article on page 27 for more details.

Third, the UTP PHY takes advantage of the two unused pairs available in every four-pair cable. Surveys of customer cable plants revealed that a large proportion of these customers adhered to structured cabling recommendations[3] when installing cable, and connected four-pair cable to each wall outlet. Two of these four pairs currently lie unused. By transmitting 5B/6B coded data at 30 megabaud on all four pairs, it is possible to provide a total signaling rate of 120 megabaud (100 Mbits/s) over UTP cable.

Quartet Signaling

The four-pair transmission scheme, called quartet signaling, uses binary transmission, that is, only two voltage levels are used as symbols. Other approaches to the UTP PMD were examined. One was multilevel (m-ary) signaling (see "Multilevel Signaling" on page 21). In a multilevel scheme, n data bits are mapped to one of $m = 2^n$ symbols, and each symbol is a unique voltage level. For example, in a quaternary scheme, two data bits may be mapped to one of four voltage levels. In this way the number of symbols transmitted, and hence the transmission rate required, is reduced by a factor of n. However, for a fixed power supply voltage, the voltage separation between symbols is reduced by a factor of $1/(m-1)$ from the binary case. The binary quartet signaling scheme maximizes the voltage separation between symbols, which provides greater immunity to noise at the receiver.

Multilevel Signaling

Multilevel signaling is often used as a means of compressing the bandwidth required to transmit data at a given bit rate. In a simple binary scheme, two single symbols, usually two voltage levels, are used to represent a 1 and a 0. The symbol rate is therefore equal to the bit rate. The principle of multilevel signaling is to use a larger alphabet of m symbols to represent data, so that each symbol can represent more than one bit of data. As a result, the number of symbols that needs to be transmitted is less than the number of bits (that is, the symbol rate is less than the bit rate), and hence the bandwidth is compressed. The alphabet of symbols may be constructed from a number of different voltage levels. Fig. 1 shows an example for a four-level scheme.

In the four-level scheme, groups of two data bits are mapped to one of four symbols. Only one symbol need be transmitted for each pair of data bits, so the symbol rate is half the bit rate. The drawback of the multilevel scheme is that symbols are separated by a smaller voltage than in the binary scheme. This means that when noise is added to the data signal (cross talk or impulse), the probability of the noise changing one symbol to another is increased. The symbol separation could be increased to that of the binary scheme by increasing the peak-to-peak transmitted voltage by a factor of (m – 1) for an m-level scheme, but this is generally not possible given fixed power supply voltages, and in any case it increases the power required for a transmitter.

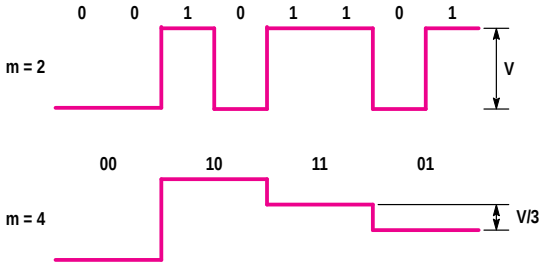


Fig. 1. Two-level and four-level signaling.

The susceptibility of a scheme to errors caused by noise is measured by the ratio of signal separation to noise. Fig. 2 shows the signal-to-NEXT-noise ratio plotted against the transmission bandwidth for several multilevel schemes and for multipair schemes for a bit rate of 100 Mbits/s. A 16-level scheme reduces the bandwidth to 25% of the bit rate, but the S/NEXT ratio is 13 dB (a factor of 4.5) worse than for a four-pair scheme with 25 Mbits/s per pair, which is the scheme used in the 100VG-AnyLAN standard.

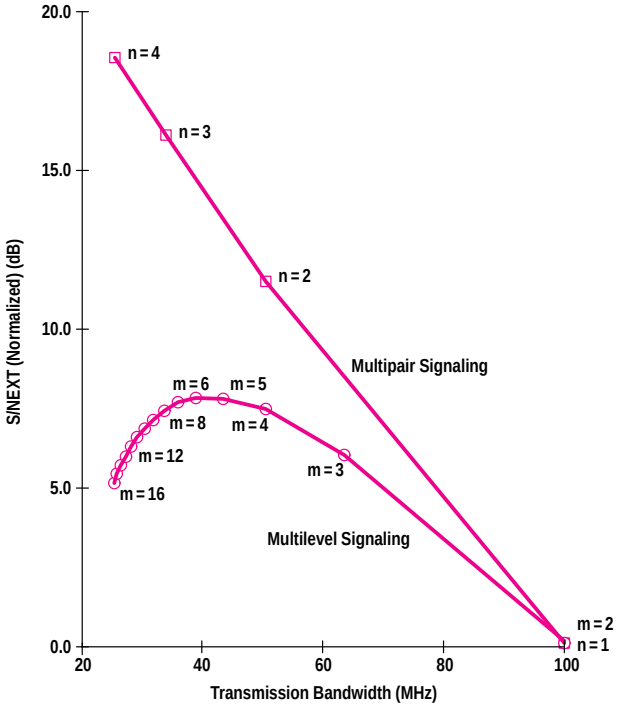


Fig. 2. Susceptibility to errors from noise.

Two noise sources are significant in UTP cabling: external noise and cross talk noise. External noise may be caused by electromagnetic radiation from radio stations (often referred to as continuous wave or CW noise) or impulses from switching equipment (impulse noise). Cross talk noise arises from capacitive coupling between twisted pairs within a cable. Cross talk affects links most significantly when the links use 25-pair bundles.

Impulse Noise

Little data is available to describe the characteristics of impulse noise occurring on UTP cable plant. It was decided that the safest approach to impulse noise was to maintain the margin of 10Base-T, since the success of 10Base-T proves that its level of robustness is appropriate. The choice of binary signaling meant that the UTP PHY could provide the same immunity to impulse noise as 10Base-T.

Cross Talk

The 100VG-AnyLAN cross talk environment is very different from that of 10Base-T. Since a four-pair UTP cable only carries a single network link in a network and the traffic on all four pairs is in the same direction, only far-end cross talk (FEXT, see "Cross Talk in Unshielded Twisted-Pair Cables" on page 19) is a problem. FEXT is less severe than near-end cross talk (NEXT) in UTP cables, and can be disregarded in four-pair links.

However, when 25-pair cables are used to connect end nodes to hubs, up to six network links (each occupying four pairs) can populate one cable. In addition to FEXT, NEXT is also a problem at the hub end of a cable. When a packet is being received on one port, retransmission of that packet at other ports will result in NEXT at the receiving point. It is essential that the level of this NEXT be minimized to prevent errors at the receiving port. The protocol minimizes this NEXT in the following way.

When a packet is received that has a single destination address, the hub forwards the packet to that destination immediately. If the source and destination are attached to the hub by the same 25-pair cable, NEXT will occur from the retransmitted packet to the received packet. An extensive analysis of the cross talk noise generated in such a scenario verified that the received signal is robust to this level of cross talk (see "Cross Talk Analysis" on page 22).

(continued on page 23)

Cross Talk Analysis

Many previous analyses of twisted-pair transmission systems have assumed that the distribution of cross talk noise is Gaussian. These have yielded reasonably accurate predictions of system bit error rates. However, in applying the error rate analysis, there is an implicit assumption that the cross talk noise is independent of the data on the disturbed system. This is often the case in telecommunication systems, but is not always the case in LANs, where the disturbing links are those on which the disturbed data is being retransmitted. For example, the NEXT interfering with data received at one port of a hub is a result of the retransmission of earlier bits of the same data on other ports. If the cross talk noise is of sufficient amplitude to cause an error in the received data, this error is extremely likely to be repeated every time the same data is transmitted. If a packet is errored by cross talk, that particular packet is likely always to be errored.

To guarantee the error-free transmission of any packet, the worst-case peak cross talk voltage must be found. The peak cross talk noise from multiple disturbers can be calculated directly from a knowledge of the cross talk channels and the disturbing data source. The NEXT and FEXT cross talk channel frequency responses can be calculated for any pair combination using measurements of the pair-to-pair NEXT or FEXT loss. Once the cross talk channel frequency response is known, it is possible to find the impulse response of this channel by inverse Fourier transform. We define the impulse response of the NEXT channel as:

$$g_n(t) = F^{-1}\left(H_n(f)\right)$$

and the impulse response of the FEXT channel as:

$$g_f(t) = F^{-1}\left(H_f(f)\right)$$

To find the cross talk noise voltage at the receiver decision point, n(t), caused by any data pattern f(t), the impulse response is convolved with f(t):

$$n(t) = f(t) \star g(t), \qquad (1)$$

where g(t) represents either $g_f(t)$ or $g_n(t)$ as appropriate. Our goal is to find the worst-case cross talk for any data pattern, so n(t) must be calculated for all values of f(t). It is therefore useful to apply some limit to the duration of f(t) to shorten the computation time, and this can be done by taking into consideration the finite duration of the cross talk channel impulse response. A typical impulse response of a NEXT channel is shown in Fig. 1. The duration of the cross talk impulse response is typically less than 1400 ns, which is equivalent to 42 symbol periods for the 30-megabaud transmission rate used in quartet signaling. Therefore, the cross talk waveform during the last six symbols of a pattern f(t) can be predicted accurately if the duration of f(t) is restricted to 1600 ns.

The cross talk for any f(t) is calculated as follows. The disturbing data source is assumed to be the output of a 5B/6B block coding function and consists of eight sequential six-bit codewords, each chosen from an alphabet of 32 codewords. The total number of permutations that f(t) can take is therefore 32^8. For each permutation, n(t) is computed according to equation 1. The peak cross talk noise voltage generated by any value of f(t) can then be found by searching each resultant n(t).

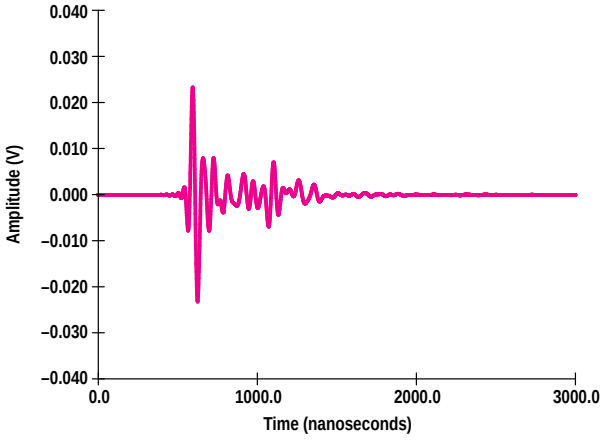


Fig. 1. Typical impulse response of a NEXT channel.

The peak cross talk noise voltage, which represents the maximum noise generated by a worst-case data pattern, can be calculated in this way for each pair combination in a 25-pair cable by repeating the search described above using the NEXT or FEXT loss particular to that pair combination. For each pair combination, the maximum cross talk noise voltage is recorded. The distribution for a typical cable is shown in Fig. 2 for NEXT and FEXT. (The cross talk noise voltage is normalized to the signal amplitude at the receiver decision point.) We denote the maximum NEXT noise voltage for pair i disturbing pair k as $v^{i,k}_{pk,NEXT}$, and the maximum FEXTnoise voltage due to pair i disturbing pair k as $v^{i,k}_{pk,FEXT}$.

Multiple-Disturber Cross Talk

The worst-case cross talk environment for UTP PMDs operating over 25-pair bundles consists of three far-end disturbers and four near-end disturbers. Modeling this environment is relatively straightforward given the distributions of $v^{i,k}_{pk,NEXT}$ and $v^{i,k}_{pk,FEXT}$. For each choice of disturbed pair (k), four NEXT disturbers (a,b,c,d) and three FEXT disturbers (p,q,r), the total noise voltage for multiple disturbers is:

$$v_{pk,total} = \sum_{i=a,b,c,d} v^{i,k}_{pk,NEXT} + \sum_{j=p,q,r} v^{j,k}_{pk,FEXT}.$$

By calculating the multiple disturber noise voltage in this way we assume, pessimistically, that the maximum noise voltages for the worst-case disturbing patterns from each disturbing source occur at the same time and with the same polarity on the disturbed pair k. This is obviously a worst-case scenario.

A Monte-Carlo approach has been used to choose combinations (k,a,b,c,d,p,q,r) randomly from the 25 pairs of a cable. For each choice, $v_{pk,total}$ was calculated. The resulting distribution of $v_{pk,total}$ is shown in Fig. 2. The maximum multiple-disturber cross talk noise expected on any pair of the cable for any choice of disturbing pairs can be estimated from the higher extreme of this distribution (such as the first percentile). This number represents the noise voltage for the worst-case choice of disturbing and disturbed pairs, with the maximum noise contributions from all disturbing pairs occurring simultaneously on the disturbed pair. For the example shown, the first percentile of the total peak noise distribution is 47% of the signal. This allows a substantial margin for error-free signal detection.

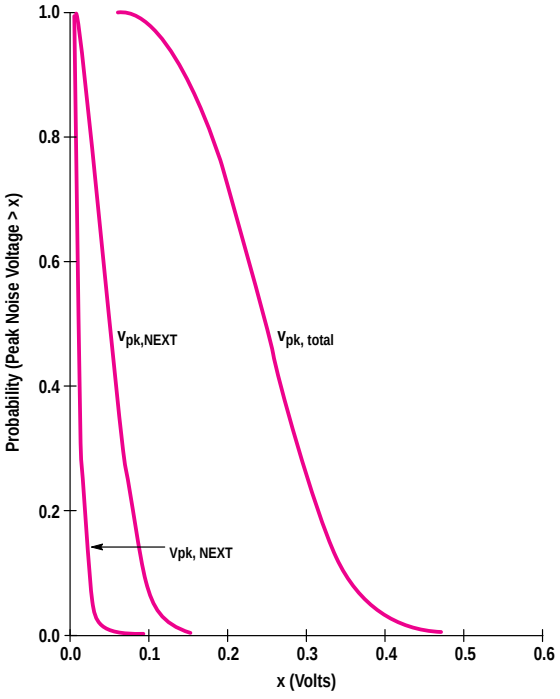


Fig. 2. Distribution of $v^{i,k}_{pk,FEXT}$, $v^{i,k}_{pk,NEXT}$, and $v_{pk,total}$ for a typical 25-pair cable.

When a broadcast or multicast packet is received, the hub does not immediately forward this packet. The NEXT caused by retransmitting the packet to several destinations sharing the same 25-pair cable as the source would result in erroneous reception. Rather, the packet is stored until reception is complete and is then forwarded to all destinations simultaneously. In this way, no NEXT occurs during reception. This store-and-forward technique is only implemented when 25-pair cables are attached to a hub. If only four-pair cables are used, all packets (single and multiple addresses) are forwarded immediately, since there is no NEXT between the individual four-pair cables.

Implementation of Quartet Signaling

Fig. 3 shows a block diagram of the 100VG-AnyLAN implementation of quartet signaling. The PMI and PMD sublayer functions are considered separately when transmitting or receiving data.

PMI Transmitting. The PMI splits the data into four streams, each of which is scrambled. This removes patterns that would result in a repetition of the same codeword in the output of the 5B/6B coder. This helps avoid spectral peaks that might violate EMC regulations when the coded data is transmitted.

Typically, LAN traffic contains data patterns that are simply repetitive 1s or 0s. If left unscrambled, when split into quintets (five bits) at the PMI, the distribution of quintets is heavily biased towards all 0s (quintet value of 0) or all 1s (quintet value of 31). This is confirmed by Fig. 4 which shows the distribution of quintets obtained from real LAN traffic. Scrambling removes this bias, providing a more random distribution of quintets at the input to the 5B/6B coder.

After scrambling, the PMI performs the 5B/6B coding. It then adds start and end delimiters to the four streams, and a preamble sequence (a 0101… pattern) to the start of each stream. The four parallel streams of coded data (30 Mbits/s per stream) are then passed to the PMD.

PMD Transmitting. The PMD converts the four parallel data streams to the binary signaling levels (±2.5V) on each of the four twisted pairs. The data is nonreturn-to-zero (NRZ) coded and low-pass filtered. The low-pass filter has a cutoff at 20 MHz, and is used to attenuate spectral components that would cause undesirable emissions above 30 MHz. A typical eye diagram at the output of the PMD is shown in Fig. 5.

PMD Receiving. When receiving data, the low-pass filter in the PMD rejects out-of-band noise on the twisted pair. The signals on the four channels are then equalized. This compensates for the attenuation of the cable and minimizes the intersymbol interference at the sampling point of the data. To perform this function for any cable length between 0 and 100 m the equalizer must be adaptive. The protocol provides a training sequence, during which the PMD equalizer trains its response to compensate for the length of cable present in the link. A typical eye diagram at the output of the receiver

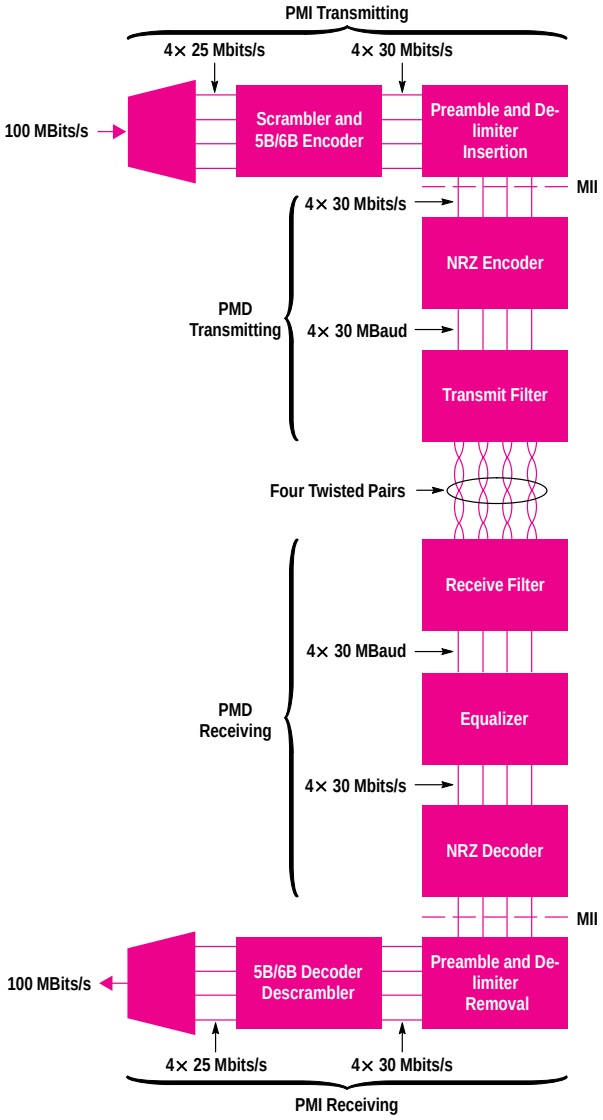


Fig. 3. Implementation block diagram. The MII is the medium independent interface.

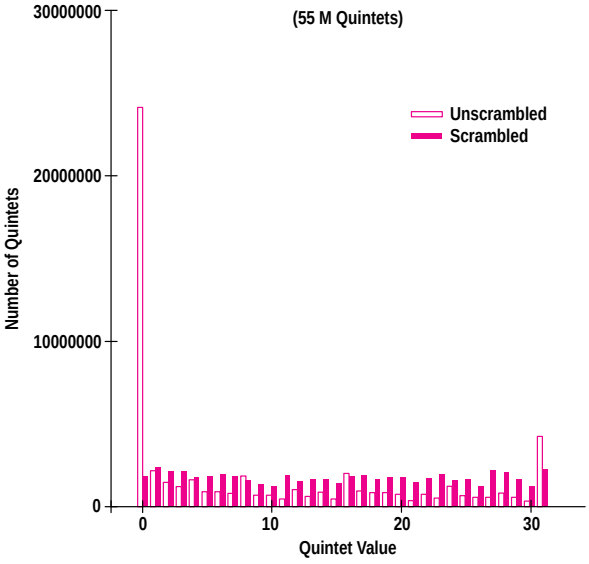


Fig. 4. Distribution of quintets obtained from real LAN traffic.

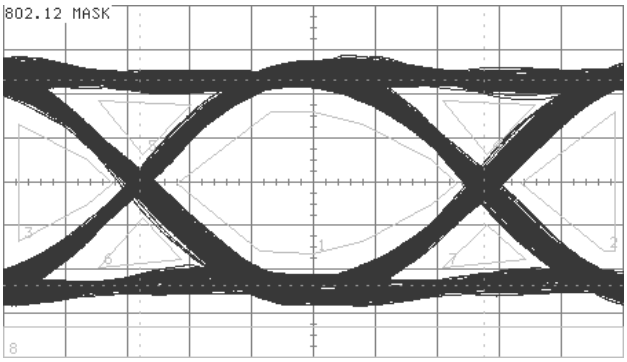


Fig. 5. Transmission eye diagram.

equalizer is shown in Fig. 6. In this case the cable length was 121 m, which approximates a worst-case 100-m cable.

After equalization the received signals are sampled. The PMD recovers a clock from the received signals by using the preamble pattern generated by the transmitting PMI. This clock is used to sample the received data. The four channels are often misaligned in time by up to two bit periods with respect to each other after traveling over a length of UTP cable. This is because the twist rate, and hence the propagation delay, varies from one twisted pair to another. The PMD realigns the received signals to a common clocking point before passing the four parallel streams of data to the PMI.

PMI Receiving. The PMI further realigns the received data to remove any skew between the start delimiters on each stream. The data on each stream is then decoded, unscrambled, and reformatted into a single data stream, which is passed to the receiving MAC. The PMI also performs a number of error-checking functions. These provide extra protection against errored packets being accepted as valid by the MAC, above the protection offered by the frame check sequence. The PMI first checks that the start delimiters on each stream are all valid and occur with the correct time relationship to each other. Then, while decoding, it checks that only valid 6-bit codewords are received. It signals to the MAC if an error is detected in the received data.

Quartet signaling meets all the design goals identified for the PHY apart from control signaling (described next). By retaining a binary signaling scheme the simplicity and robustness of 10Base-T was maintained. Keeping the baud rate low makes it easy to implement in a standard CMOS process, thereby meeting the low-cost requirement. The increased data rate is mainly attributable to more efficient block coding and better use of the available twisted-pair channels.

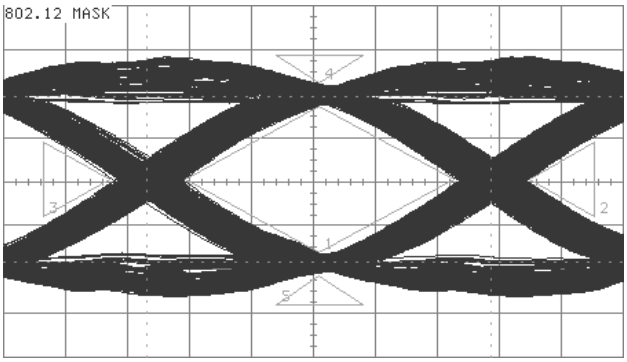


Fig. 6. Reception eye diagram.

Control Signaling

The protocol uses control signals to transfer requests, acknowledgments, and training signals between the hub and the end nodes. These control signals must be continuously available on all network links whenever data is not being transferred. They must be full-duplex and distinct from any data pattern.

Since the control signals are required to operate continuously, signaling at the data rate of 30 megabaud was rejected because of the large levels of cross talk that might be generated when 25-pair cables were used. Instead, a low-frequency signaling scheme has been developed that allows all five control states to be transmitted using only signals with fundamental frequencies less than 2 MHz. No valid data pattern contains components with such low frequencies, providing distinct identity for the control signals. Full-duplex operation is achieved by using two pairs in each direction to carry control signals.

Two control signals are defined: CS1 is a 0.9375-MHz square wave formed by repeating a pattern of sixteen 0s followed by sixteen 1s, and CS2 is a 1.875-MHz square wave formed by repeating eight 0s followed by eight 1s. Four of the five control states are represented by combinations of CS1 and CS2 on two pairs as shown in Table I.†

The fifth control state is represented by silence (no energy) on the two pairs. Silence is used by a hub to indicate that a request to transmit data has been granted. On receiving silence, an end node can cease control signaling and begin transmission immediately, since the cable is already silent. This allows rapid turnaround of the half-duplex link.

The two square waves have been chosen to have a large separation in frequency so that the receiver is able to distinguish them without having a clock that is precisely phase-locked. The control signals are generated and recovered within the UTP PMD.

The STP and Optical-Fiber PMDs

The STP PMD was developed to support existing token-ring network cabling. These consist of cables up to 100 m in length with two shielded twisted pairs. The optical-fiber PMD was developed to provide a means of connecting hubs and end nodes over longer distances than the 100 m provided by the STP and UTP PMDs. This extra distance is particularly important when cascaded networks are built and hubs are distributed over a campus area. The optical-fiber PMD allows LANs to have up to 4-km diameter.

† The control signals shown in Table I are for the 100VG-AnyLAN implementation of the IEEE 802.12 standard. The control signal definitions in the standard have different names.

Table I
Control Signaling in 100VG-AnyLAN

Tone Wires		Transmitted from			Received by		
1	2	End Node	Root Hub	Another Hub	End Node	Root Hub	Another Hub
CS1	CS1	IDLE	IDLE	IDLE	IDLE	IDLE	IDLE
CS1	CS2	REQ_N	INCOMING	REQ_N	INCOMING	REQ_N	INCOMING
CS2	CS1	REQ_H	ENABLE_HIGH_ONLY	REQ_H	Reserved	REQ_H	ENABLE_HIGH_ONLY
CS2	CS2	REQ_T	REQ_T	REQ_T	REQ_T	REQ_T	REQ_T

REQ_N = Normal-priority request
REQ_H = High-priority request
REQ_T = Link training request
INCOMING = A packet is about to be transmitted
ENABLE_HIGH_ONLY = Put normal round-robin sequence on hold while a high-priority request is serviced

STP has less attenuation and greater NEXT loss than even Category 5, data-grade UTP, and is already used for data transmission at 100 Mbits/s. For example, the SDDI specification[4] provides for transmission of FDDI traffic at 100Mbits/s over STP.

The STP physical layer uses the same modulation and coding techniques as SDDI, that is, binary signaling at a rate of 100 Mbits/s (before coding). The low attenuation of the cable makes it possible to transmit 100Mbits/s on a single pair, so one pair is dedicated to transmitting and one pair to receiving at each end of a link. The cable shield reduces radiated emissions to satisfactory levels.

Multimode optical-fiber links have also been in use for many years at data rates of over 100 Mbits/s (such as FDDI).[5] The optical-fiber PMD uses components that have recently been developed for low-cost FDDI implementations. Examples are the HFBR 5106 and HFBR 5107 developed by HP's Optical Communications Division. These use LED transmitters with wavelengths of either 850 nm (for 500-m links) or 1300 nm (for 2-km links). The transmitter forms part of a small module that also includes an optical receiver. This module forms a simple interface between the optical fiber and a transceiver chip, which can be identical to the STP transceiver (see "Optical-Fiber Links for 100VG-AnyLAN" on page 26).

For the demand priority STP and optical-fiber PHYs, the block code and NRZI coding specified for SDDI have been replaced with an alternative scheme based on the same 5B/6B block code that is used for the UTP PHY. This approach allows the 5B/6B block coder to be placed in the PMI sublayer. The amount of logic that is common to all physical layers is thereby increased, resulting in a lower-cost PHY device.

The PMI provides four 30-Mbit/s channels of scrambled and 5B/6B block coded data at the MII. These four channels are multiplexed, codeword by codeword, by the STP and fiber PMDs (see article, page 27). Even after codeword multiplexing, the 5B/6B code retains its advantageous properties. The serialized data stream is NRZ coded and transmitted at 120 megabaud on one pair using symbol levels of ±0.25V for STP media or passed to the optical module mentioned above for optical-fiber cables.

Control Signaling on STP and Optical Fiber

The two-pair, two-tone control signaling developed for the UTP PMD is not suitable for the STP and optical-fiber PMD because only one pair per direction is available for control signaling. Because cross talk is not an issue with STP or optical-fiber links, we were able to explore the use of higher-frequency signals. Square waves were again attractive because they can easily be chosen to be distinct from valid data patterns.

One concern was that the control signals not produce harmonics that, when transmitted on STP, might cause radiated emissions that violate regulations. It was decided that the control signal spectra should always fall below the random data spectrum, since it was known that the data transmission met EMC regulations. Calculations showed that square waves with frequencies less than 4 MHz met this requirement. A lower bound on the control signal frequency was set by the need to avoid distortion of the square wave resulting from transmission through the transformers used to couple to twisted-pair.

The final choice of control signals is five square waves with frequencies between 1.875 and 3 MHz. Again, the control signals are separated in frequency sufficiently to allow detection without a phase-locked clock at the receiver.

Summary

The physical layer developed for 100VG-AnyLAN local area networks accommodates different cable types by means of three different physical medium dependent (PMD) sublayers.

The major goal was to provide 100-Mbit/s transmission on the existing cable plant. The three PMDs allow LANs to operate across the vast majority of LAN media installed today: UTP (categories 3, 4, and 5), STP, and multimode fiber. As a result, customer investment in structured cabling is protected while at the same time an upgrade path to high-speed LANs is created. All three PMDs are designed to be simple and cost-effective. Customers can now benefit from a factor of up to eight improvement in the cost/performance ratio of their LANs without the significant cost penalty of replacing their cable plant.

Optical-Fiber Links for 100VG-AnyLAN

As data rates increase, low-cost optical-fiber links play an increasingly significant role in LANs for extending the length of links beyond what can be achieved with copper media, while meeting the full range of electromagnetic emission and susceptibility requirements for networks.

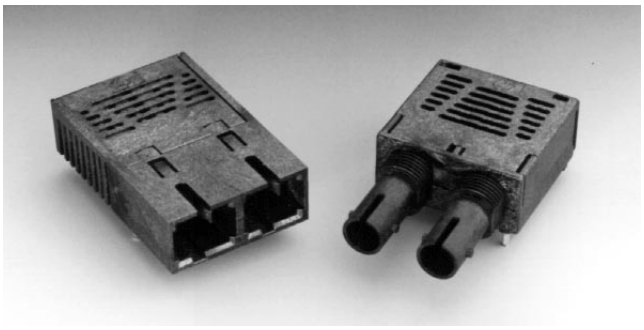


Fig. 1. HP optical transceiver.

The 100VG-AnyLAN standard defines a serialized interface with a 120-megabaud signaling rate for STP and multimode optical-fiber links. The standard defines two optical-fiber link length specifications, which allow the use of low-cost 850-nm technology for 500-m building backbones (this is the same technology used in the existing IEEE 802 standard CSMA/CD 10Base-F and 802.5J token-ring links) and 1300-nm technology for 2-km campus backbone links.

Fig. 1 shows the new Hewlett-Packard low-cost industry-standard optical transceiver package. This small package is 1 inch wide and 1.5 inches long and has a duplex SC optical connector on the front and a 1-by-9 row of electrical pins at the rear. HP transceivers HFBR 5106/5107 meet the two 100VG-AnyLAN link length standards and allow interchanging link technology in the same printed circuit board footprint. The transceivers are also available with AT&T ST optical connectors to address the large installed base having ST connectors for building and campus backbones.

Del Hanson
Principal Engineer, Fiber-Optic
Networks and Standards
Optical Communications Division

Acknowledgments

The authors were fortunate to be part of an enthusiastic and effective team based at HP Laboratories Bristol and the Roseville Networks Division. Major contributions to the physical layer developments were made by Pat Thaler of RND. Rod Chidzey of HP Laboratories was responsible for the construction of the cable cross talk measurement rig and for gathering cable data. Del Hanson of the Optical Communications Division provided valuable assistance in the specification of the fiber-optic PMD.

References

1. E.A. Lee and D.G. Messerschmitt, *Digital Communication, Second Edition*, Kluwer Academic, 1994.

2. *Twisted-Pair Medium Attachment Unit (MAU) and Baseband Medium, Type 10Base-T*, IEEE Standard 802.3, section 14.4.4.1.

3. *The Commercial Building Telecommunications Wiring Standard*, EIA/TIA Standard 568, July 1991.

4. *SDDI: A Specification for FDDI on Shielded Twisted-Pair*, implementors' agreement published by AMD Inc., Chipcom Corp., IBM Corp., Madge Networks Inc., Motorola Inc., National Semiconductor Corp., Network Peripherals, Inc., Sumitomo Corp., Synoptics Communications Inc., SysKonnect Inc., and Technitrol Inc.

5. *Fiber Distributed Data Interface (FDDI)—Token Ring Physical Layer Protocol (PHY)*, ANSI Standard X3.148-1988.

Coding in 100VG-AnyLAN

A 5B/6B coding scheme in which five data bits are encoded into six-bit codewords is used in conjunction with offsetting the data on different channels by three bits in quartet signaling. It provides the level of error detection necessary, produces a signal balanced within narrow limits, and restricts strings of consecutive 0s or 1s to a maximum length of 6. It is also efficient.

by Simon E. C. Crouch and Jonathan Jedwab

Coding of data before transmission has three main purposes. First, it can ensure that the transmitted signals are dc balanced, that is, that there are equal numbers of 0s and 1s over an extended period of time. This is important in transformer-coupled systems, which are designed to prevent ground loops. Second, coding can aid clock synchronization by minimizing the length of strings of consecutive 0s or 1s. This ensures that there is a high density of signal transitions. Third, together with cyclic redundancy checks (CRCs), it can enable errors to be detected. There is an IEEE Project 802 functional requirement that any three bits in error within a data frame must be detected. There is also a need to detect burst errors; any coding system must not compromise the detection provided by the CRC.

The 100VG-AnyLAN team chose a 5B/6B coding scheme in which five data bits are encoded into six-bit codewords. This is used in conjunction with offsetting the data on different channels by three bits in quartet signaling. It provides the level of error detection necessary, produces a signal balanced within narrow limits, and restricts strings of consecutive 0s or 1s to a maximum length of 6. At the same time, it is efficient, adding only 20% to the data load, compared with the 100% produced by the 1B/2B Manchester coding used in 10Base-T. The 5B/6B code is also effective when the four data channels are multiplexed for transmission on STP or fiber-optic cables.

This article describes the design of the 5B/6B block code used in 100VG-AnyLAN and more generally in the IEEE 802.12 proposed local area network standard[1] and explains the reasons behind its design.

Why Code?

The method used to transmit digital data has to take account of the physical constraints that the transmission medium imposes upon the transmission system. The physical media used by 100VG-AnyLAN include unshielded twisted pair (UTP) cable, shielded twisted pair (STP) cable, and optical fiber. The demands imposed by the use of UTP cable mean that to transmit at 100 Mbits/s (even when using quartet signaling), digital data has to be coded before it is transmitted. The design of the code has to take account of various constraints imposed by implementation considerations and by the need to be compatible with other data transmission systems. In particular, the 5B/6B 100VG-AnyLAN code scheme was designed to be compatible with the packet structure and error detection capabilities defined by the IEEE 802.3 standard used in Ethernet local area networks and by the IEEE 802.5 standard used in token ring local area networks.

Block Codes and Error Detection

Generally speaking, a code is a mapping from an alphabet of symbols to a set of sequences of symbols from some other alphabet. So, for example, we could define a code from the alphabet {a,b,c,...,z} to the set of sequences of zeros and ones by: a $\rightarrow$ 010, b $\rightarrow$ 0110, c $\rightarrow$ 01110, and so on. For another example, the representation of decimal numbers as their binary equivalents can be regarded as a code. This idea of a code should not be confused with the notion of a cipher, which is a code that is specifically designed to hide the meaning of transmitted data from those not meant to understand it. The codes that we are using are designed to be easy to decode and are capable of detecting errors in transmission. For the mathematical background of coding theory, see reference 2. For an engineering perspective, see reference 3.

A block code is a specific type of code in which every codeword has the same length. In a 5B/6B block code, each element of an alphabet of 32 different symbols (which itself can be represented by a 5-bit number) is encoded as a 6-bit codeword.

The major physical constraint that affects the use of a coding scheme is the need to maintain dc balance, that is, the spectral content of the transmitted signal should have no zero-frequency component. In mathematical terms, this translates into requiring that the difference between the number of ones and the number of zeros transmitted at any time must be kept as close to zero as possible.

Clearly, a simple way to get a balanced code is to represent a data character of 0 by the codeword 01 and a data 1 by the codeword 10. The number of 1s transmitted will then equal the number of 0s transmitted at the end of each codeword. Unfortunately, this is also a very inefficient way of achieving dc balance because it requires the physical transmission of two code bits for every data bit sent. The 100VG-AnyLAN team chose to implement a 5B/6B code because it gives a suitable trade-off between efficiency and cost of implementation.

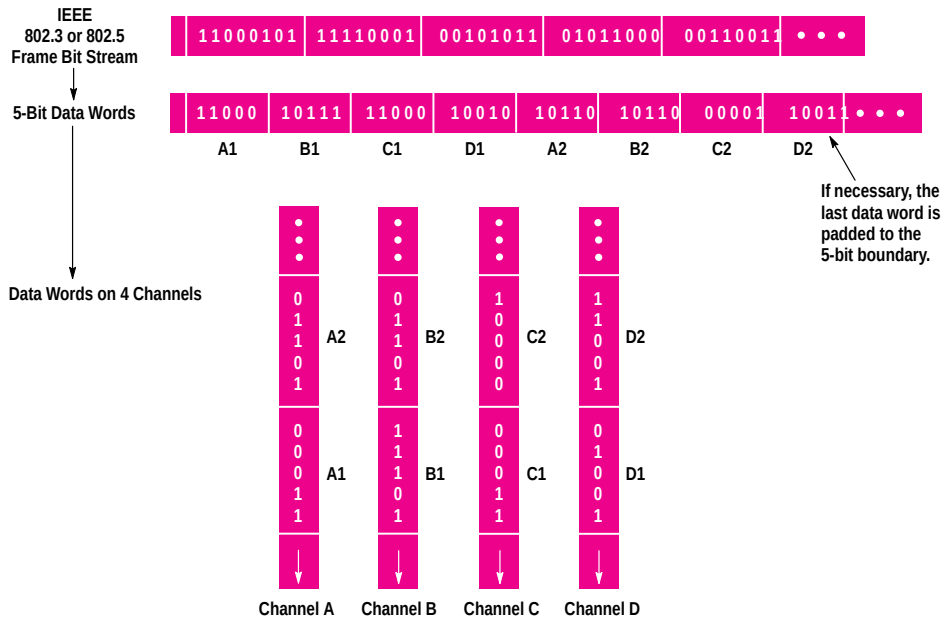


Fig. 1. Distribution of a data stream on four channels for quartet signaling.

The major compatibility constraint imposed upon the design of the 100VG-AnyLAN 5B/6B code was that it had to be compatible with the MAC frame formats defined by the IEEE 802.3 and 802.5 standards and in particular with the error-detection capabilities offered by the cyclic redundancy check (CRC) built into those formats. (See "Polynomial Arithmetic and Cyclic Redundancy Checks" on page 31). CRC-32, the CRC defined by both IEEE 802.3 and 802.5, detects up to three single-bit errors occurring anywhere in a frame, or a single burst error of up to 32 bits in the frame. The protocol will then discard that packet as flawed. (A data packet consists of the MAC frame delimited by a preamble and start and end delimiters.) The 100VG-AnyLAN team had to design the 5B/6B code so that similar error-detection characteristics will be maintained when 802.3 or 802.5 frames are coded and transmitted in parallel using quartet signaling. In particular, the requirement to detect up to three single-bit errors anywhere in a frame is a compulsory requirement of IEEE Project 802,[4] so the new IEEE 802.12 standard had to meet this demand.

In the next two sections we describe the process of coding the data frame and the error detection capabilities of 100VG-AnyLAN.

Coding a Data Packet

Data from the IEEE 802.3 or 802.5 frame is divided into 5-bit data blocks (with padding added at the end, if necessary) and is distributed between the four data streams transmitted using the quartet signaling scheme (see Fig. 1).

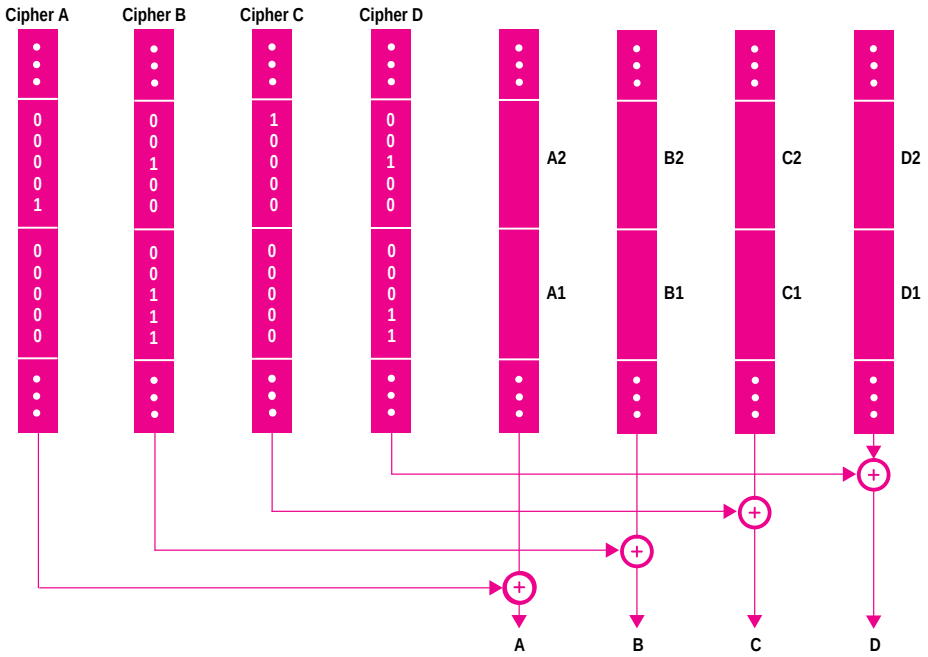


Fig. 2. The data words are XORed with pseudorandom stream ciphers.

Each data stream is exclusive-ORed with a stream of pseudo-random bits produced by stream ciphers (see Fig. 2). The ciphered data blocks on the four streams are then coded according to the following 5B/6B code table, as shown in Fig. 3.

Data	Code	Data	Code Pair
00001	101100	00000	001100 110011
00011	001101	00010	100010 101110
00101	010101	00100	001010 110101
00110	001110	01011	000110 111001
00111	001011	01100	101000 010111
01000	000111	01110	100100 011011
01001	100011	10000	000101 111010
01010	100110	10010	001001 110110
01101	011010	10101	011000 100111
01111	101001	10111	100001 011110
10001	100101	11010	010100 101011
10011	010110	11110	010010 101101
10100	111000		
10110	011001		
11000	110001		
11001	101010		
11011	110100		
11100	011100		
11101	010011		
11111	110010		

In the second column, all the 6-bit codewords are balanced, that is, there are equal numbers of 0s and 1s. There are not enough balanced 6-bit codewords to code all possible 5-bit data symbols, so twelve data symbols are coded by a choice of two different 6-bit codewords, one of weight two (two 1s and four 0s), and one of weight 4 (four 1s and two 0s). The two codewords are used in the following fashion, independently in each stream:

- For the first data symbol that codes to an unbalanced codeword, the weight-two codeword is chosen.
- When the next unbalanced codeword occurs, the weight-four codeword is used.
- Weight-two and weight-four codewords continue to alternate whenever an unbalanced codeword occurs.

The observant reader will notice that in all but one of the code pairs, the alternative unbalanced codewords are logical negatives of each other. This is not true of the second pair, for reasons having to do with the error properties of the code.

At the end of the stream, one of two end delimiters is used, independently in each stream (see Fig. 4). When the end of the stream is reached, if the next unbalanced codeword is due to have weight 2 (according to the rules above), end delimiter ED2 is used. If it is due to have weight 4, ED4 is used. The invalid packet marker (IPM) is used by repeaters to mark errored packets for disposal or further processing.

After coding, the data streams are offset by 3 bits with respect to each other before transmission (see Fig. 3). Again, the reason for this has to do with error-detection properties and will be explained below.

The actual transmitted bit sequence down each channel looks as shown in Fig. 4. The start and end of packet markers are designed to maintain error detection capabilities. The use of alternative end delimiters (effectively an extra parity check) is an essential part of the error detection scheme of 100VG-AnyLAN.

Multiplexing

Another design constraint of the 5B/6B coding scheme was that it should behave well when the four data streams are multiplexed onto fewer channels for networks using STP or fiber-optic cables. As noted in the article on page 18, STP and fiber-optic PMDs pass four parallel streams of data through a multiplexer, which combines the four codeword streams into one stream, codeword by codeword (Fig. 5).

When combined in this way, the physical and error protection capabilities of the 5B/6B code are maintained, as explained in the next section.

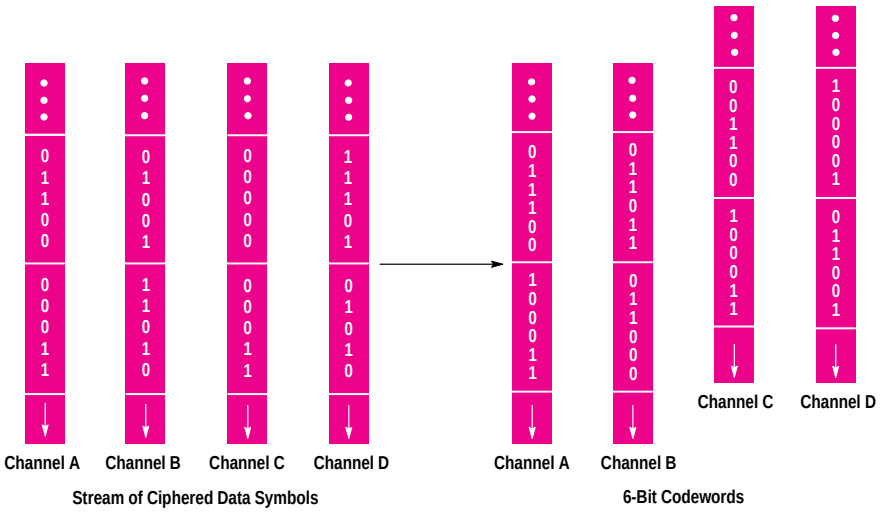


Fig. 3. Conversion of 5-bit data symbols to 6-bit codewords. The three-bit offset of channels C and D is for error detection.

IEEE 802.3 and 802.5 Frame Formats

The IEEE 802.3 frame (Fig. 1) consists of a 48-bit destination address, a 48-bit source address, a 16-bit length field, and then a data field ranging between 368 and 12,000 bits in length and consisting of data organized into octets. This is followed by a 32-bit cyclic redundancy check (CRC).

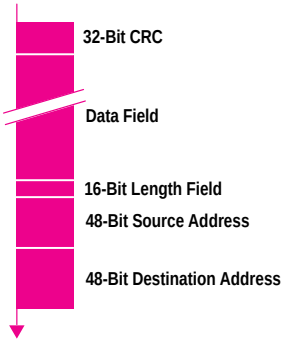


Fig. 1. IEEE 802.3 frame structure.

The IEEE 802.5 token-ring frame (Fig. 2) consists of an 8-bit access control field (not used in the 802.12 standard), an 8-bit frame control field, a 48-bit destination address, a 48-bit source address, between 0 and 240 bits of routing information, and then a data field ranging between 0 and 36,016 bits in length and consisting of data organized into octets. This data field is followed by a 32-bit CRC.

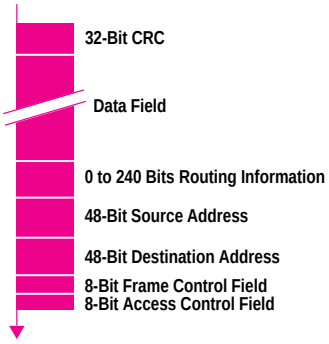


Fig. 2. IEEE 802.5 frame structure.

The CRC provides two types of protection for an IEEE 802.3 or 802.5 frame:

- Any three single-bit errors occurring anywhere in the frame are detected.
- Any burst of errors for which the distance between the first corrupted bit and the last corrupted bit is less than or equal to 32 bits is detected.

IEEE Project 802 requires that any transmission scheme developed under its aegis can perform the first type of error detection, that is, that at least three single-bit errors can always be detected.

Properties of the Coding Scheme

The choice of 5B/6B code, together with the alternation of unbalanced codewords and the use of the alternative end delimiters, ensures that IEEE 802's requirement is met: up to three single-bit errors occurring anywhere on the four channels within a data frame are detected. This is not a trivial matter to confirm, because the use of a code means that a single-bit error in the transmitted bit stream may cause many more than one single bit to be in error in the data stream after decoding. A substantial portion of the code

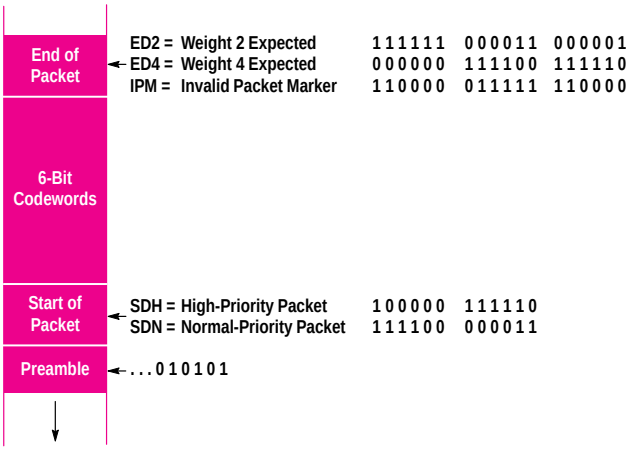


Fig. 4. Transmitted bit sequence on each channel.

design effort was dedicated to limiting the number of data bits affected by a single code bit in error.

It is not difficult to see how the use of the alternation rule for choosing unbalanced codewords together with the use of the alternative end delimiters leads to the detection of all triples of single-bit errors, when those errors occur in distinct codewords on a single channel. Fig. 6 shows several cases in which three single errors are detected by the alternation rules and the use of end delimiters.

A second major design thrust was to make sure that burst error-detection properties inherited from CRC-32 are not compromised in 100VG-AnyLAN. Because we are now talking about a system that can transmit over four parallel streams as well as a single stream, we have to be careful to define what we mean by a burst error. We decided that the clearest case could be made by considering a burst error to be any error caused by arbitrary corruption across all the streams for a certain number of bit periods (see Fig. 7). So, for example, for a burst error of seven code bit periods, there is a block seven code bits long by four code bits wide (corresponding to the four channels), within which any given code bit may or may not be inverted.

The first thing we realized was that if all the codewords were transmitted synchronously on all four channels, a burst error of length 2 (corrupting eight code bits) could cause trouble if it occurred at a codeword boundary (see Fig. 8). The error could corrupt eight code words and thus eight data blocks, corresponding to forty data bits. This is well above the 32 protected by the CRC. It would therefore require a remarkably good coding scheme to protect against even such a short error burst.

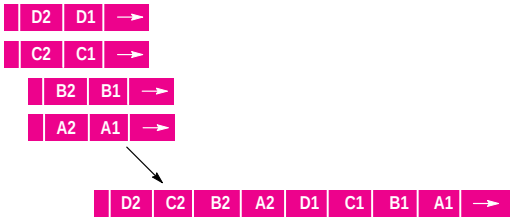


Fig. 5. Multiplexing four streams into one.

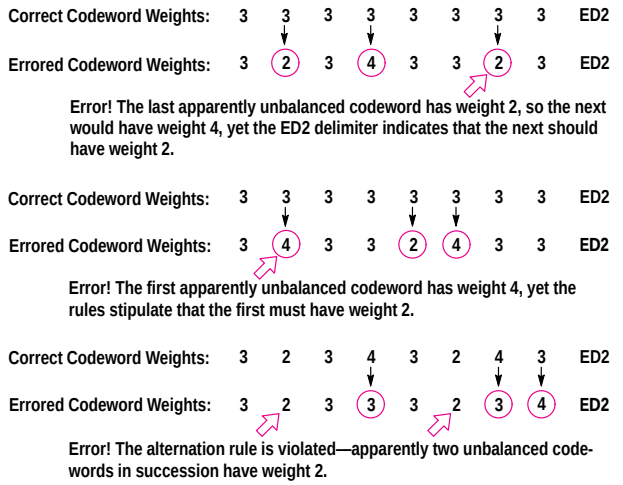


Fig. 6. Detection of errors through the alternation rule and end delimiters.

The solution to this problem is to offset the codewords on two of the channels relative to the other two. Fig. 9 clearly shows that with such an offset, even an error burst of length four (corrupting 16 code bits) can only corrupt the code bits of at most 6 codewords. This corresponds to 6 data blocks (30 data bits) and so will be detected automatically by the burst error properties of the CRC. Thus, even with an arbitrary code, any burst error of length four is detected.

Substantial theoretical work led us to design a code on top of this offset scheme that extends the burst error-detection capability to seven code bit periods (arbitrarily corrupting 28 code bits), while still meeting the requirement to detect three separate single-bit errors (see Fig. 10). The triumph of theory over trial and error should not be underestimated here—there are approximately 10^{45} choices of 5B/6B code!

When multiplexed onto a single channel for STP or fiber-optic solutions, the error detection capabilities of the 5B/6B code are enhanced. The single-bit error protection remains at three while the burst error-detection capabilities are increased to 34 code bits.

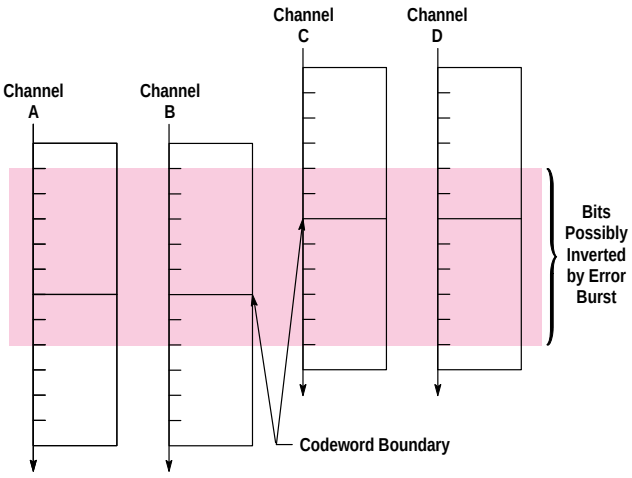


Fig. 7. Illustration of a burst error with a length of seven code bit periods.

Polynomial Arithmetic and Cyclic Redundancy Checks

The calculation of cyclic redundancy checks (CRCs) depends upon the arithmetic of modulo 2 polynomials. A modulo 2 polynomial is an expression of the form $a_0 + a_1x + a_2x^2 + ...+ a_nx^n$, where the coefficients a_0, a_1, ... ,a_n are integers modulo 2, that is, they can take values of 0 or 1. The coefficients obey regular modulo 2 arithmetic as follows:

Addition

+	0	1
0	0	1
1	1	0

Multiplication

×	0	1
0	0	0
1	0	1

Electrical engineers will recognize the addition operation as the XOR operation defined on binary states.

The following are examples:

- $(1+x^2+x^4) + (1+x+x^4) = x+x^2$
- $(1+x^2+x^4)\ (1+x+x^4) = 1+x+x^2+x^3+x^5+x^6+x^8$.

Division of modulo 2 polynomials is done in exactly the same way as it is for ordinary polynomials, remembering that the coefficients obey modulo 2 arithmetic as defined above.

A very important operation in the calculation of CRCs is the calculation of the remainder R(x) when one polynomial M(x) is divided by another G(x). R(x) is uniquely defined by M(x) = G(x)K(x) + R(x), where the degree of R(x) is less than the degree of G(x). The degree of a polynomial is defined as the largest value of n for which the coefficient a_n is nonzero, so, for example, $1+x^2+x^5$ has degree 5.

Although this sort of polynomial division may look formidable, there are very efficient means of calculation based on shift registers.

CRCs are calculated on a stream of data by assuming that the data represents the coefficients of some modulo 2 polynomial. So, given a stream of data n bits long, the first bit can be considered as the coefficient a_{n-1} of x^{n-1}, the second bit as the coefficient a_{n-2} of x^{n-2}, the $(n-1)$th bit as the coefficient a_1 of x and the nth bit as the constant term a_0.

Roughly speaking, the 32 bits of the CRC are defined to be the polynomial remainder R(x) when the polynomial defined by the data M(x) is divided by a standard polynomial:

$$G(x)= 1+x+x^2+x^4+x^5+x^7+x^8+x^{10}+x^{11}+x^{12}+x^{16}+x^{22}+x^{23}+x^{26}+x^{32}.$$

Slight modifications are made for implementation reasons, but as far as the error properties are concerned, this is what is calculated.

The CRC bits are appended onto the data. When the data is received, the CRC is calculated and compared with the received CRC. If there is a difference, the data is known to have been corrupted.

When a corruption occurs in transmission, a number of bits are inverted. Let the errored bits define the coefficients of a polynomial E(x); for example, if errors occur at bit positions p and q, the polynomial will be $x^{n-p} + x^{n-q}$. Since the operation of calculating CRCs is linear, the error is detected if and only if E(x) is not exactly divisible by G(x). CRC-generating polynomials such as G(x) are chosen precisely to detect as many polynomials like E(x) as possible.

In addition to the error detection, the 5B/6B code has other properties that are highly pertinent to physical transmission. The first of these is run length—the maximum number of consecutive zeros or consecutive ones transmitted on any code stream. This is important in transmission systems where some form of clock information is recovered from the data stream, since clock recovery usually depends on receiving a

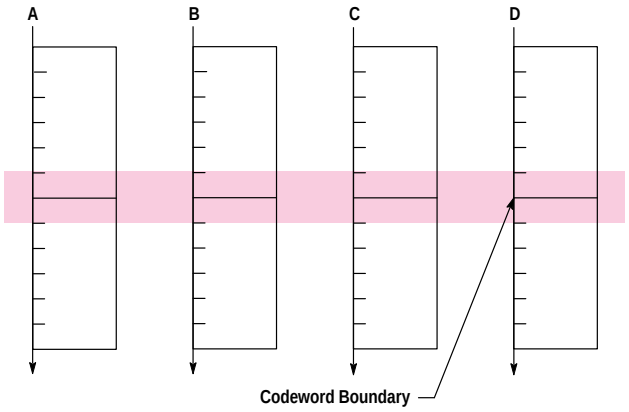


Fig. 8. A burst error across synchronous channels. A burst of length 2 corrupts up to eight code bits.

reasonable density of signal transitions. For the 100VG-AnyLAN 5B/6B code this maximum run length is 6, whether on four channels or on one.

Another physical attribute of the 5B/6B code is the running digital sum (RDS)—the difference between the number of zeros and the number of ones in the transmitted bit stream since the code transmission began. A bounded RDS ensures bounded baseline wander. For the four-pair scheme, the RDS is bounded between -5 and $+3$. For the single-stream scheme, the bounds are -11 and $+3$. The bounds are not symmetrical because of the alternation rule—a weight-2 codeword is always sent first on a channel when there is a choice of unbalanced codewords

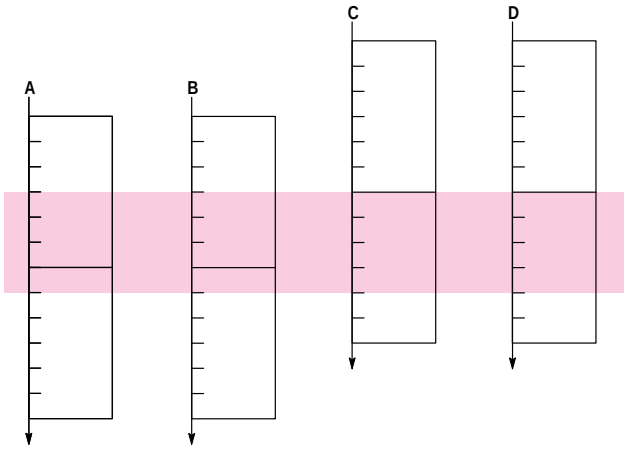


Fig. 9. Detection of a burst of length 4. A burst of length 4 will always be detected because of the offset between channels.

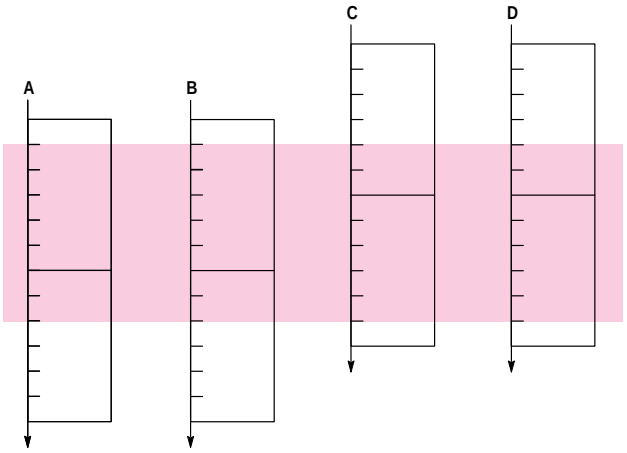


Fig. 10. Detection of a burst of length 7. A burst of length 7 is always detected because of the offset and the coding scheme.

In summary then, the choice of 5B/6B code, with the alternative end delimiters working on top of the offset transmission scheme, allows 100VG-AnyLAN to meet IEEE 802's requirements. It will detect three separate single-bit errors anywhere in the frame and is capable of detecting substantial burst errors for all the media choices while maintaining favorable physical transmission attributes.

Acknowledgments

The authors would like to thank all of the team members at HP Laboratories Bristol and the Roseville Networks Division for their individual contributions, which made the team effort so fulfilling. Special thanks should go to Pat Thaler and Dan Dove at RND and to David Cunningham, Alistair Coles, Miranda Mowbray, David Coker, and Michael Spratt at HP Laboratories, who all provided valuable advice in their own special ways.

References

1. *Information Technology—Local and Metropolitan Area Networks—Part 12: Demand Priority Access Method and Physical Layer Specification*, IEEE, 1994.

2. J.H. van Lint, *Introduction to Coding Theory*, GTM 86, Springer Verlag, 1982.

3. R.E. Blahut, *Theory and Practice of Error Correcting Codes*, Addison Wesley, 1983.

4. *Functional Requirements, IEEE Project 802*, Local and Metropolitan Area Networks Standards Committee, Draft 6.10.

Multimedia Applications and 100VG-AnyLAN

Networks must guarantee bandwidth for multimedia traffic and must control end-to-end delay and delay jitter (fluctuation in the arrival time of packets). The new campus network, 100VG-AnyLAN, can meet these requirements in many circumstances through the basic operation of the protocol. More flexibility can be obtained through the use of bandwidth allocators and the target transmission time protocol. Until either the Broadband Integrated Services Digital Network (B-ISDN) or reliable Internet protocols become available, the use of dial-up remote bridges with existing WANs can accommodate multimedia traffic in the near term.

by John R. Grinham and Michael P. Spratt

The term *multimedia* covers a wide range of applications, ranging from simple image-in-text word processors through video conferencing. The qualifier "real-time" confines the applications to those that impose timing constraints on the underlying delivery system, such as:

- Conferencing systems using synchronized video, audio, and shared computer data
- Remote training and information databases containing digital video and audio data
- Remote control or manipulation of computer-generated models and robotic machines.

As desktop computers grow in power and decrease in price, and networks expand both in area and in data transfer rates, such applications are likely to have a major impact on all desktop computer systems and workgroup LANs in the mid to late 1990s. The factors that indicate a takeoff in multimedia can be divided into two groups: user demand and the development of appropriate technology.

User Demand for Multimedia Applications

A major demand is for humans to interact with remote systems. This started in the fields of education and training. Computer-based training, which used to use analog video and audio, has become a mature market, and is moving rapidly towards using digital media. In the future, this data may need to be held centrally so that it can be updated easily. Training applications could be networked to this central source. This requires that control data be transmitted to a server with low delay, and that the server have the ability to transmit multimedia data back to the desktop, also with low delay.

Applications are growing in the scientific, engineering, and medical domains in which, for example, humans interact in real time with remote models and image data. These applications also place significant timing constraints on the underlying network.

Perhaps the most exciting growth area is in the support of real-time human conversation. Teams separated geographically need such systems for activities traditionally performed on a single site, such as meetings, training, marketing, and support. As an example, HP has already released a set of collaborative workgroup tools that includes imaging, digital audio, and shared data.

Systems such as these are a natural development from video conferencing systems. It is remarkable that over half of all Fortune 1000 companies use boardroom-based conferencing facilities, given that until recently they cost over $20,000 per node and were relatively crude in operation. The possibility of low-cost desktop-based conferencing systems that integrate with existing computer applications has to be seen as extremely attractive to companies and teams distributed over several sites.

Multimedia Technology

Demand for desktop multimedia applications connected over networks can only be satisfied if the technology exists to support it. Three areas are critical:

- The development of standards for video and audio coding that allow low-cost implementation and fast transmission
- The development of low-cost wide-area networks (WANs)
- The development of consistent interfaces to multimedia technology, both from WANs to appropriate LANs, such as 100VG-AnyLAN, and from LANs to applications.

Video and Audio Coding Standards. The ISO and the ITU-T (formerly CCITT) have produced several standards for audio and video coding algorithms, such as MPEG and H.261 for video conferencing and video telephony and JPEG for the compression of still images (now being extended to full-motion video). Since all these algorithms are based on discrete cosine transformations, low-cost chipsets that support all three are being developed by several manufacturers.

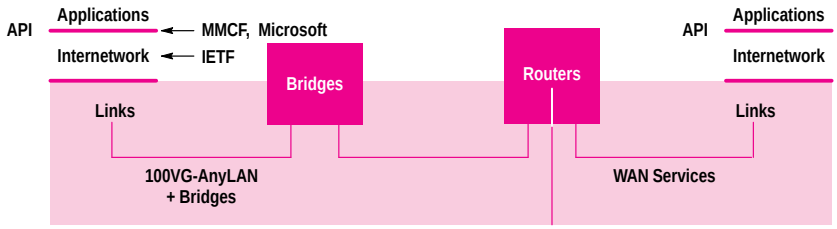


Fig. 1. Developments in different portions of a wide area network.

Audio CD-quality systems are already becoming standard features of workstations and PCs, and many manufacturers have released plans to incorporate hardware support for full-motion video. The increasing power of workstation and PC microprocessors is also making software compression and decompression of full-motion video feasible. HP's current workstations can already perform decompression of full-motion video. Real-time compression is likely to follow in a few years.

Wide Area Networks (WANs). The cost of WAN services has fallen dramatically in the last few years. For example, a long-distance video conferencing link that was around U.S.$1500 per hour in 1984 now costs in the range of U.S. $12 to $30 per hour. In addition, there is increasingly widespread availability of a variety of dial-up WAN services.[1]

There is also a significant effort going into the development of wide area broadband services, perhaps leading to a Broadband Integrated Services Digital Network (B-ISDN) by the end of the decade.

Consistent Interfaces. For multimedia to work effectively over wide area networks and a variety of campus and local-area networks, there is a need for consistent interfaces between networks. 100VG-AnyLAN only covers a portion of a wide area networked system.

Various bodies are looking at other areas of network systems. Fig. 1 illustrates the situation schematically.

The Multimedia Communications Forum (MMCF), the Winsock 2.0 forum, and the Integrated Services Group of the Internet Engineering Task Force (IETF) are looking at network interface issues, while the IMA has been considering end-system services. Most major operating system and hardware manufacturers, including HP, have been involved in addressing these areas, and are developing standards to ensure consistency of interfaces.

Demands of Multimedia on Networks

There are three areas in which real-time multimedia applications make demands on networks:

- Workgroup LAN delay must be minimized.
- It must be possible to determine delay bounds for all elements in the transfer path from source to destination so that it can be guaranteed that data will arrive at the destination within a specified time.
- Delay jitter (fluctuations in the arrival time of packets) must be controlled.

The following sections describe these three considerations in more detail and indicate how 100VG-AnyLAN addresses them.

Control of End-to-End Delay. The most important new requirement of real-time multimedia applications is for control of end-to-end delay. [2,3,4] The ITU-T's target is a maximum of 150 ms for low-quality voice services. Real-time 3D visualization transmissions are more critical. Even one-way delays of more than 45 ms have been found to cause unpleasant side effects. We decided to set a target of about 100 ms for WAN applications and 30 ms for local applications.

Table I illustrates the contribution to delay of various elements in the data path of a remote collaboration system. The system is shown in Fig. 2.

Table I
Delays in a Remote Collaboration System

System Component	Typical Delay Value
Video Coder Delay	2 to 150 ms, depending on the CODEC used
Video Decoder Delay	2 to 150 ms, depending on the CODEC used
Packetization Delay	8 ms, using 1500-byte packets and a 1.5-Mbit/s stream
Workgroup LAN Delay	0.1 to over 200 ms, depending on the protocol used
Backbone Network Delay	10 ms typical for each backbone traversed
Router/Bridge Delay	5 to 10 ms, if priorities are implemented
Wide Area Network Delay	5 ms per 1000 km, or about 25 ms for the U.S.A., coast to coast

Of these delays, video coding and decoding have the greatest ranges. CODEC hardware is evolving extremely rapidly. Those using a JPEG processor such as the C-Cubed Microsystems CL550 have a delay of less than 10 scan lines (a few milliseconds). Currently H.261 and MPEG CODECs can impose a considerable delay. These should reduce as the technology progresses, but even so, it is possible for the

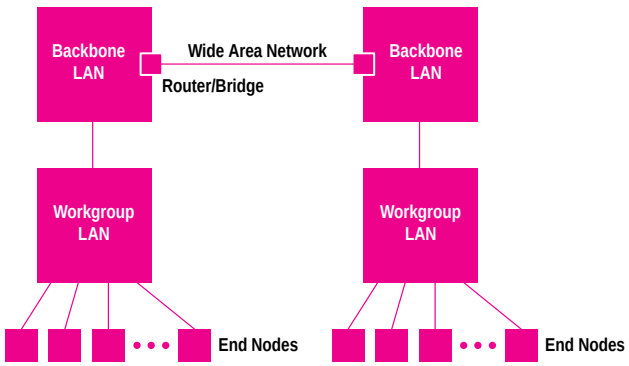


Fig. 2. Example of a remote collaboration network.

CODEC, packetization, and wide area network delays to account for all of the 100-ms delay budget. It follows then, that for effective transmission of real-time multimedia, the workgroup LAN delay must be minimized. To achieve our targets for wide area and local transmissions, we set a workgroup LAN delay target of about 10 ms.

In many cases the basic operation of the 100VG-AnyLAN protocol keeps packet delay across the LAN bounded below 10 ms. As network size and complexity increase, the high priority level can be used for multimedia traffic, possibly in conjunction with resource allocation schemes as described on page 37.

Bandwidth Guarantees. Application vendors need to be confident that their applications will work no matter what other traffic is on the network. The need for guaranteed high-bit-rate services to a single node or a few nodes extends across all elements within a network complex, from the LAN to bridges and routers and wide area networks. One simple method of connecting LANs to a WAN is through the use of a dial-up remote bridge (see "Remote Bridge Example" at right).

Within a LAN, 100VG-AnyLAN can guarantee bandwidth to multimedia applications either by exploiting the basic operation of the protocol or, if more flexibility is required, through the use of resource allocation schemes as described on page 37.

Interarrival-Time Jitter. Packet switching has been put forward as an effective technology for integrating stream-type data, such as video and audio, with conventional computer data. Packet switching can allow flexible allocation of bandwidth to individual calls, interfaces that are well-matched to computer systems, and efficient multiplexing of sources.

However, packet switching introduces a problem. Packets will be subject to variations in arrival time (known as interarrival-time jitter). How then can reconstruction of the continuous audio or video signal from a set of packets sent over a network be ensured?

Jitter is produced when a source transmits data packets at regular intervals but each packet is subject to a different delay before it arrives. Delay is caused partly by fixed elements, such as the propagation time and packetization delays, and partly by variable delays from queuing and waiting for access to the network. It is these last two that result in jitter.

Provided that any jitter is kept below a maximum value, it is possible to calculate a buffer size at the receiver that is sufficient to smooth out any variation in interarrival times. The data can then be retrieved from the buffer in a steady stream for decoding and display. Fig. 3 shows how this is achieved when an arrival window can be determined.

The arrival window must be small, or the buffering delay will cause the total end-to-end delay of the transmitted data to exceed the 100 ms target.

Remote Bridge Example

To illustrate the operation of a dial-up remote bridge, Fig. 1 shows two sites, each with a backbone FDDI network linking 100VG-AnyLAN hubs.

Users connected to end nodes 1 and 2 want to communicate by video and audio between the two sites. The backbone networks on each site are connected to the wide area network (WAN) through dial-up remote bridges specifically designed to carry multimedia traffic. We assume that site A and site B can also communicate through existing routes, such as some kind of corporate internet.

When the call between node 1 and node 2 is initialized, an isochronous link with the appropriate bit rate is set up between the two dial-up bridges. Bridge A is told the 48-bit MAC address of node 2. Any subsequent packets that appear on the backbone FDDI bearing that destination address will be forwarded across the WAN. Bridge B puts them on the backbone FDDI at site B, from which they reach node 2.

Any normal-priority data packets destined for node 2 will not be picked up by bridge A because they will carry the destination address of the router connecting site A to the corporate internet, not that of node 2.

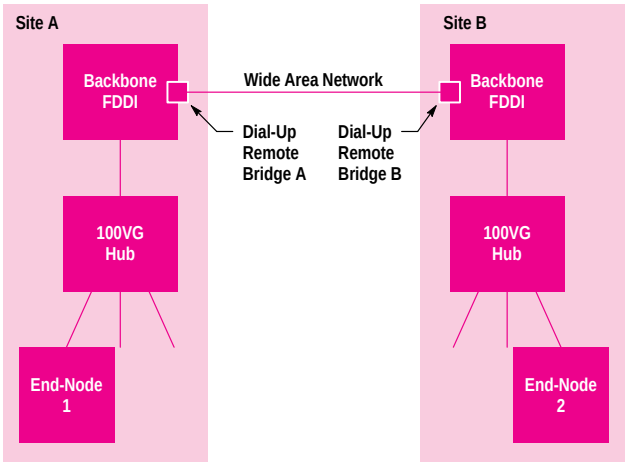


Fig. 1. Remote sites connected by dial-up remote bridges.

Introducing New Applications

Because interface standards and technology are still being developed for the whole wide area network structure, it is

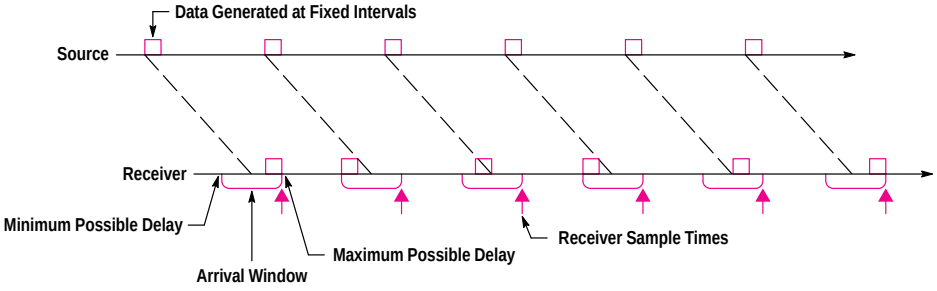


Fig. 3. Recovering jittered data.

Higher-Level Protocols

The introduction of new LAN technologies (such as 100VG-AnyLAN) is only one element in the development of multimedia networking. Higher-level protocols are also needed that can control and transfer stream data over multiple network hops. The Internet Engineering Task Force (IETF) and the International Telecommunications Union (ITU-T) have been working in this area.

IETF

The IETF has been developing protocols for a multiservice, packet-based network spanning the world (see Fig. 1), such as the Internet Stream Protocol (ST-II). This is a network-layer protocol roughly equivalent to Internet Protocol (IP), but specifically developed to support stream-based traffic. ST-II is being used in the Multimedia Teleservices section of the BERKOM-II program (see "Related Projects" on page 38).

The IETF has a number of groups looking at new protocols for multimedia that can be introduced into the worldwide Internet.

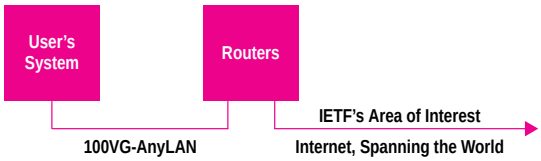


Fig. 1. IETF involvement in worldwide networks.

ITU-T Narrowband ISDN Standards

The ITU-T works towards developing standards for data transmission over an enhanced telephone network, such as ISDN or other digital circuits with fixed bit rates up to 2 Mbits/s (Fig. 2). 100VG-AnyLAN could provide desktop connections to these services. ITU-T Recommendation H.320 includes the following:

- H.261: Video CODEC for audiovisual services at $p \times 64$ kbits/s
- H.242: A system for establishing communication between audiovisual terminals
- H.221: A frame structure for a 64-to-1920-kbit/s channel in audiovisual teleservices
- Q.931/2: A D-channel signaling prot col.

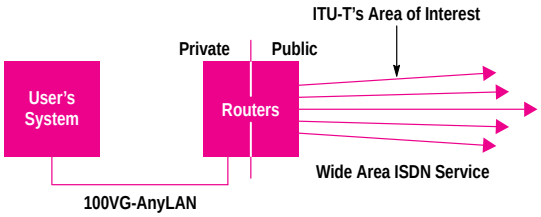


Fig. 2. ITU-T involvement in worldwide networks.

important that there be an easy growth path for the introduction of new multimedia applications. This also satisfies the need for users with existing networks in place to be able to build them up to cope with advanced multimedia traffic in stages, using the existing infrastructure and avoiding a large initial outlay. Unlike other high-speed campus LANs, it is possible with 100VG-AnyLAN to provide such a growth path for the introduction of multimedia applications.

The growth path has three phases:

- Phase 1. Run the application using a single priority level throughout the network.
- Phase 2. Move some nodes to high priority as multimedia traffic increases, with simple mechanisms for monitoring and controlling bandwidth.
- Phase 3. Introduce automatic bandwidth allocation mechanisms for all high-priority traffic.

Phase 1: Multimedia Using a Single-Priority 100VG-AnyLAN.

In the first phase, since no new API for applications has been developed, the application is simply put on normal priority all the time. Two of the properties of the 100VG-AnyLAN demand priority protocol enable it to be used with multimedia applications even in this phase: bounded delay and bandwidth sharing.

Bounded Delay. The access delay for a data packet is guaranteed to be less than or equal to the maximum packet duration times the number of active ports on the hub. If Ethernet-length packets are used, the maximum packet length is 12,000 bits, giving a duration of 0.12 ms. With a 32-port hub, a queued packet will have to wait at most $0.12 \text{ ms} \times 32 = 3.84$ ms. This will increase to 7.84 ms if all traffic is multicast and the system is using bundles.

These figures compare with $32 \times$ TTRT (target token rotation time) for asynchronous FDDI (giving 320 ms for a 10-ms TTRT), and unbounded for CSMA/CD.

Bandwidth Sharing. Every active node has an equal share of transmission opportunities. In the worst case, when every node is actively transmitting full-length packets of data at the same priority, this will give a maximum bandwidth of about 3 Mbits/s per node on a 32-port hub. This reduces to 1.5 Mbits/s if all data is multicast in bundles. This guarantee is more than sufficient for many applications such as simple conferencing.

It follows that a 32-port hub can support simple applications using real-time data at 1.5 Mbits/s or less, and application vendors can be confident that the system will work no matter what other data is on the LAN.

Phase 2: Use of High Priority without Allocation

If the second priority is used, the two properties listed in Phase 1 apply to high-priority traffic. If 32 nodes run at high priority in a large network, say of 100 nodes, the guarantees would be the same as in the previous example.

A malfunctioning high-priority node can be prevented from saturating the network by network monitoring (the hub can refuse access to a high-priority transmission), and by configuring the network so that normal-priority traffic is raised to high-priority after waiting a certain period of time.

This provides a simple way of introducing multimedia applications, but in the longer term, a mechanism for allocating bandwidth may be required. This will control the use of bandwidth by high-priority traffic automatically, and will ensure that delay is kept within an upper bound on large networks.

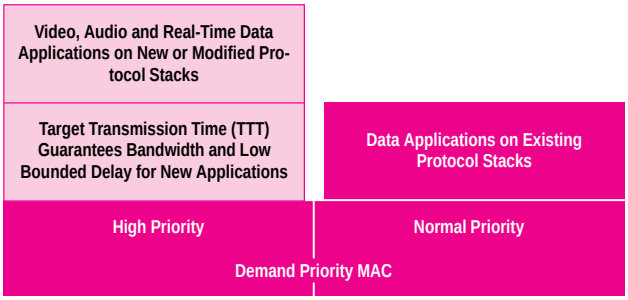


Fig. 4. 100VG-AnyLAN end node stacks.

Phase 3: Resource Allocation in 100VG-AnyLAN

A third property of 100VG-AnyLAN facilitates the allocation of bandwidth on networks: spare bandwidth utilization. Any bandwidth not used by high-priority traffic is immediately available for normal-priority data.

This property is particularly relevant if high-priority traffic is bursty. It also ensures that normal-priority traffic does not need to participate in the allocation process.

Fig. 4 illustrates how the protocol stacks for high and normal priority are independent. This is important because resource allocation protocols need only be introduced on end systems that actually run multimedia applications. End systems that only use normal priority do not need any new software. Note that the use of a bandwidth allocator requires higher-level interfaces and services to be defined, such as those on which the IETF is working (see "Higher-Level Protocols" on page 36.

Example of Bandwidth Allocators. Consider the following simple scenario illustrating the use of a bandwidth allocator. Several 100VG-AnyLAN hubs are connected by an FDDI ring. The allocator ensures that the sum of the high-priority synchronous multimedia connections across the FDDI does not exceed 80 Mbits/s, and that the sum of the high-priority multimedia connections across any individual 100VG-AnyLAN network does not exceed 80 Mbits/s. This ensures that the 100VG-AnyLAN networks and the FDDI backbone are not overloaded with high-priority traffic.

Suppose that all the 100VG-AnyLAN hubs are 32-port or smaller. The protocol guarantees that in the event of all nodes simultaneously wishing to transmit normal-priority data, they will each have a bandwidth of a little over 3 Mbits/s. The allocator then only has to ensure that the sum of the high-priority synchronous multimedia traffic across the FDDI does not exceed 80 Mbits/s. A system such as this can typically support up to 80 video connections each with a peak bandwidth of 1 Mbits/s, or up to 200 364-kbit/s connections.

In 100VG-AnyLAN, the proposed mechanism for allocating bandwidth and setting delay bounds is the target transmission time.

Example of Target Transmission Time Allocation. A nominal time value, the target transmission time (TTT) is set, and nodes request permission from the allocator to transmit a maximum amount of data within a TTT period. Provided that the sum of all such requests is less than a preset maximum, say 80% of the network capacity, the allocator grants the request. The end node is then guaranteed that its data will be transmitted during the TTT period.

Fig. 5 shows a simple example of how this could work. In the example, there are four stations operating at high priority. Nodes 1 and 2 are each generating 30 Mbits/s, node 3 is generating 7.5 Mbits/s, and node 4 1.5 Mbits/s.

Node 4 generates one 1500-byte packet every 8 ms and requires this to be transmitted onto the network within 8 ms. If this is the lowest access delay, the TTT would be set at 8 ms. In this time period, the other nodes will generate 30kB (20 1500-byte packets), 30 kbytes, and 7.5 kbytes (5 1500-byte packets) of data, respectively.

Fig. 5 shows how the data in each of the end nodes' transmission queues is transmitted on the network.

Bandwidth Monitoring. High-priority traffic can be monitored centrally at the hub or through a separate device attached to a promiscuous port. From packet lengths and interpacket gap times, the monitor can easily calculate the aggregate

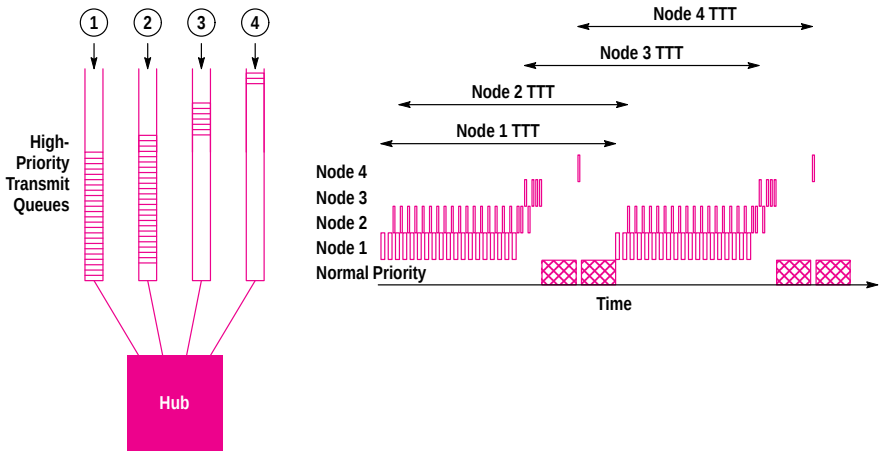


Fig. 5. Example of the use of the target transmission time (TTT).

Related Projects

Two projects are under way that have a bearing on the development of networks to carry multimedia traffic.

Teleservices Project

The Teleservices project is part of the Deutsches Bundespost (DB) Telekom BERKOM technology and demonstrator program. The program covers wide area network and campus broadband communication studies, using broadband switches and local optical-fiber links installed in Berlin to form a large test bed. It now forms part of the German Broadband Integrated Services Digital Network (B-ISDN) asynchronous transfer mode (ATM) pilot network, with switching centers in Cologne and Hamburg.

The Teleservices Project is investigating the provision of multimedia services over wide area, broadband networks. HP is involved in two areas: multimedia collaboration (MMC), supporting conferencing and shared applications, and multimedia transport (MMT), developing high-speed networking protocols for multicast operation.

HP joined in April 1993, with collaboration from HP Germany, HP Laboratories in Bristol, and TUB (Technical University of Berlin, as an HP subcontractor). The main HP/TUB task is the development of high-speed network protocols and the overall implementation and integration of the complete collaboration application under the multimedia collaboration task for the demonstration of interoperability. Five HP 9000 Model 755 computers have been installed at the TUB for the multimedia implementation, together with a range of audiovisual equipment, communications interfaces, and associated software packages. Prototype HP ATM switches have also been installed.

The program will produce a limited HP multimedia communications and management product for use on HP 9000 Model 7x5 workstations in the German broadband trials. It will also provide measurement and performance evaluation results, including:

- Observations of the interactions between video and audio compression schemes, transport protocols, the different network technologies used, and their impact on perceived performance
- Identification of particular issues and problems associated with multimedia applications related to HP products and commercial interests
- Performance measurement and behavior of a transport protocol designed for multimedia applications running over different network technologies.

A demonstration was given in August 1993 using the first version of the conference application with software video compression and relatively simple control options. In March 1994, an enhanced version was demonstrated at the CeBIT exhibition in Hannover. This had a range of management and control options, an audio server, and full-motion hardware compression (M-JPEG). It also included multivendor terminal interworking at the show over the ATM demonstration network and conferencing over ATM link connections back to a terminal in Berlin.

A demonstration of the Teleservices package between HP Laboratories in Bristol and the TUB, using ISDN at about 300 kbits/s showed that a minimum transfer rate of 1 Mbit/s is required for successful operation. Wideband operation over ISDN connections is also being evaluated.

Distance Learning Seminar

In October 1994, a pan-European seminar on distance learning was held on a network linking three European universities (the Royal Institute of Technology, Stockholm, the Technical University of Berlin, and University College London) and various other European sites. The seminar was sponsored by HP and aimed to promote the use of distance learning and to explore the future of the technology and the role of multimedia.

The seminar was a practical demonstration of multimedia over a mix of transmission technologies. German Telekom provided B-ISDN (broadband ISDN ATM) links between Berlin and London and Berlin and Stockholm. Other links used inverse multiplexers at each site to provide contiguous bandwidth over ISDN of 384 kbits/s. This provided adequate bandwidth for 25-frames-per-second video using hardware compression based on the H.261 standard, plus good-quality audio.

throughput of the hub for each TTT. If the aggregate is higher than a certain threshold value, further action can be taken on individual streams. For example, the network manager could disable high-priority traffic at a particular port, or this could be done automatically through the network monitor. A technique such as this provides a high level of security to other nodes, even from malicious attempts to disrupt the high-priority service.

References

1. D. Blankenhorn, "MCI Announces Switched T1 and T3 Services," *Newsbytes*, 1991.
2. "General Characteristics of International Telephone Connections and Circuits, Recommendations G.100-G.181," *ITU-T Blue Book*, Vol. III, Fascicle III.1, 1988.
3. A. Shah, D.Staddon, I. Rubin, and A. Ratkovic, "Multimedia over FDDI," *LCN Conference Proceedings*, 1992, p. 110.
4. I. Corbett, "Videophony," *Audiovisual Telecommunications*, BT Telecomms Series, Chapman & Hall, 1992.

100VG-AnyLAN 15-Port Hub Design

Much of the intelligence and uniqueness of a 100VG-AnyLAN network is concentrated in the hub. Special repeater, transceiver, and end node chips implement the functionality of the HP J2410A AdvanceStack 100VG Hub 15.

by Lisa S. Brown

Hewlett-Packard has introduced several network products for 100VG-AnyLAN applications. Products available since mid-1994 include a 15-port hub, a 10Base-T/100VG bridge and SNMP management module, a 10Base-T/100VG ISA adapter card, and a 10Base-T/100VG EISA adapter card. These products were developed before completion of the IEEE 802.12 standard and are currently under modification to include changes that occurred in the standard between the time of product definition and the final version of the standard. Most notably, this includes changes to support token ring framing. Additional 100VG-AnyLAN products available in 1995 are a 10Base-T/100VG PCI adapter card, a multiport hub, a fiber hub, and a 10Base-T/100VG switch.

Much of the intelligence and uniqueness of a 100VG-AnyLAN network is concentrated in the hub. Therefore, this article mainly focuses on the design of HP's 15-port hub, the HP J2410A AdvanceStack 100VG Hub 15 (see Fig. 3 on page 9). Some 100VG-AnyLAN-specific aspects of the other products are also explored in this article.

Hub Architecture

The key components of a 100VG-AnyLAN hub are the repeater chips and transceivers. In the HP 15-port hub solution, a microprocessor subsystem provides configuration and management functions. Fig. 1 is a block diagram of the 15-port hub.

The repeater chip implements the demand priority protocol, the signaling to receive and transmit packets, and the interfaces to the system. The system interfaces include a port

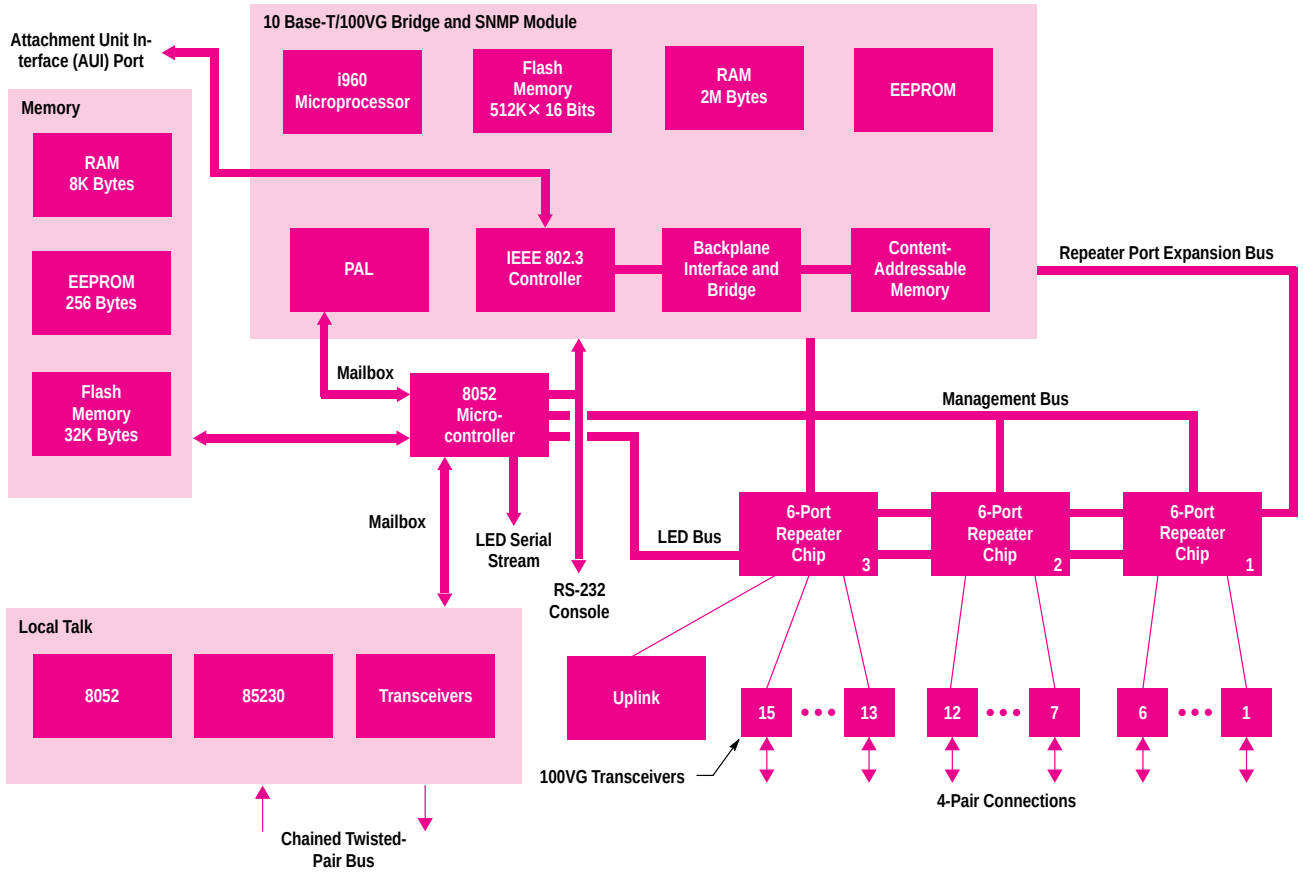


Fig. 1. Block diagram of the HP J2410A AdvanceStack 100VG Hub 15, a 15-port 100VG-AnyLAN hub.

expansion bus to connect several repeater chips into one logical repeater and an interface for network management functions, such as control and status information. The repeater chip is a six-port device that connects directly to six transceivers. Additionally, it has a single uplink port which also connects to a transceiver for use in a cascaded hub topology. For the 15-port hub, three repeater chips provide the 15 ports plus one downlink. The repeater chips act as one logical repeater through the use of the parallel expansion bus.

The transceiver chip implements the physical layer as described in the article on page 18. It is used in conjunction with a filter specially designed for use in 100VG-AnyLAN applications. The 15-port hub uses 16 transceiver chips, one for each of the 15 ports plus another for the uplink connection. The transceivers connect directly to the repeater chips.

Repeater Chip

The repeater chip logic was developed by HP and AT&T engineers in conjunction with the IEEE 802.12 standard development. It is now commercially available through AT&T as the ATT2R01. It implements the following functions:

- Demand priority protocol
- 100VG-AnyLAN RMAC (repeater MAC)
- Port expansion bus to connect multiple repeater chips together
- Interface to six transceivers for downlink ports
- Interface to transceiver for uplink port
- Packet buffer RAM
- Six-port CAM (content-addressable memory)
- Management bus
- Control and status registers
- Management counters.

Fig. 2 is a block diagram of the repeater chip's internal modules.

A packet traverses through the hub in the following manner. The protocol state machine polls the ports as described in the article on page 13 and determines which port will receive a packet from its end node next. It initiates the proper tone sequence through the transceiver and readies the hub to receive a packet. When a packet is received, it first passes through the receive portion of the 100VG repeater MAC. The data is decoded as described in the article on page 27 and is passed through as bytes. Statistics for network management functions are gathered on the packet data and the packet is passed through to the port expansion bus. Each repeater chip on the port expansion bus loads the packet into its own internal packet buffer. The packet buffer is implemented as a RAM that is large enough to hold a maximum-length packet. The first six bytes of the packet are sent to the CAM. At this point, the destination address of the packet is decoded and compared with all entries in the CAM. The transmit control circuit decides which ports, if any, will transmit the packet to an end node. This is based on whether there was a match with a CAM entry, if the packet was a broadcast or multicast packet, or if any of the ports are configured as promiscuous. Finally, the packet passes through the transmit portion of the 100VG repeater MAC where it is encoded for transmission as described in the article on page 27.

The repeater chip also implements the training functions as described in the article on page 13. The protocol state machine controls the training sequence in a manner similar to its control of data transfers. As the training packets are received, the chip learns the status of the connected port and writes the source address into the internal CAM. It monitors the received packet, and if an error is detected the packet is not counted as a good training packet. Training continues until 24 sequential good packets are transmitted in each direction (see "Invalid Packet Marker" on page 41). If the

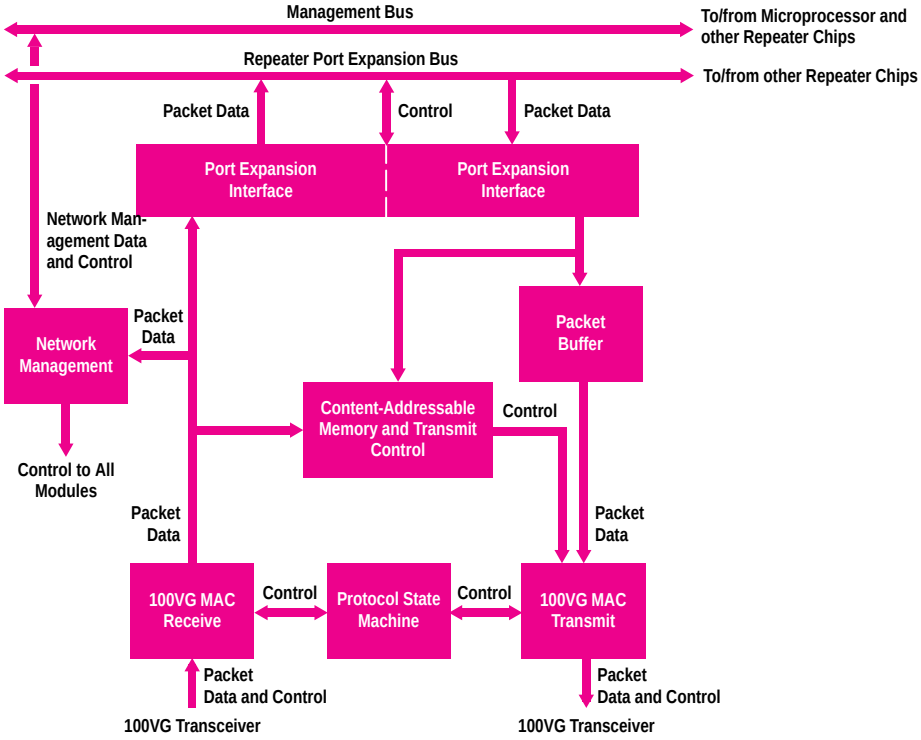


Fig. 2. Repeater chip block diagram.

Invalid Packet Marker

When a packet has been received with an error, the invalid packet marker is appended when the packet is retransmitted to an end node or another hub in a cascaded network. Invalid packet marker information is useful during training and during normal packet transmission.

During training, if a packet is received at the end node with an invalid packet marker attached, this implies that an error occurred on the link where the packet was transmitted from the end node to the hub. If a packet is received at the end node with no invalid packet marker but an error is detected on the link, this implies that an error occurred on the link where that packet was transmitted from the hub to the end node. In either case, the packet is not counted as a good training packet.

During normal packet transmission, the detection of the invalid packet marker indicates that an error occurred in a link upstream from where the invalid packet marker was detected. For example, in a cascaded repeater network, the combination of receive error and invalid packet marker detection can be used to determine which link in the cascade produced the error. Excessive errors on any one link could indicate a marginal physical connection. A network administrator might then decide to examine the link to improve network performance.

link is faulty, the node will not pass training and the protocol state machine will pause the training attempt, wait one to two seconds, and then allow training to be attempted again.

The management bus allows the processor subsystem to access the repeater chips. In the network management block of Fig. 2 there are many control registers, which network management applications can set as needed. These include functions such as disabling ports, allowing for promiscuous ports, setting the hub in a bundled cable configuration, security functions, and a host of others. The processor also has read access to the CAM, which can aid network managers in building a map of the network. Additionally, there are sixteen 24-bit counters per port and a number of status registers in the network management block that can be read via the management bus. The counters accumulate statistics on each port such as the number of bytes received, good and bad packets received, errors, high-priority and normal-priority packets received, and others. The LED control interface is a simple bus that provides port status information to the 8052 microcontroller for driving the hub LEDs.

Transceiver Chip

The transceiver chip logic was developed by HP and AT&T engineers in conjunction with the IEEE 802.12 standard development. It is now commercially available through AT&T as the ATT2X01. It implements the following functions:

- 100VG-AnyLAN physical layer (PHY)
- Quartet signaling
- Adaptive equalization
- Timing recovery
- Data justifier
- Receive control signals (RCS) machine
- Transmit control signals (TCS) machine.

Further information on the functionality of the transceiver chip can be found in the article on page 18.

Network Interface Card Architecture

The key components in the network interface cards, such as the 10Base-T/100VG ISA and EISA adapter cards, are the 10Base-T/100VG end node chip, the external buffer RAM, the 100VG transceiver, and the 10Base-T transceiver. The 10Base-T transceiver is available commercially through a number of vendors and is used to provide a 10-Mbit/s interface to Ethernet networks. The 100VG transceiver is the same device that is used in the hub, the ATT2X01, and is described above. The 10Base-T/100VG end node chip is a new device to accommodate both 10-Mbit/s and 100-Mbit/s networks.

The 10Base-T/100VG ISA and EISA adapter cards are implemented with the same end node chip, which includes both the ISA and EISA system buses and the 100VG and 10Base-T interfaces to transceivers. An EISA workstation adapter card is available for HP 9000 Series 700 servers. The end node chip was developed by HP and AT&T engineers and is now commercially available from AT&T as the ATT2MD01.

The end node chip implements ISA and EISA bus interfaces specifically tuned to handle 100 Mbits/s in the most efficient manner possible. The EISA interface provides bus mastering to allow 100VG servers to achieve LAN link rates with minimal CPU overhead. These bus interfaces comply with the industry-standard specifications for ISA and EISA machines.

Each end node device implements a 100VG module, which consists of a 100VG MAC and a protocol state machine. Through direct connection to a 100VG transceiver, this module generates and recognizes the appropriate tone sequences for training and data packet transmission. It converts internal chip bus data to 5B/6B encoded data and it decodes incoming 5B/6B data for the internal chip bus.

The end node chip also implements an interface to a 10Base-T network. It complies with all IEEE 802.3 specifications and connects directly to commercially available 10Base-T transceivers. This interface allows customers who may still be installing 10-Mbit/s networks to upgrade easily to 100VG-AnyLAN in the future without replacing adapter cards.

Other System Components

HP has also implemented a combination 10Base-T/100VG bridge and SNMP module. The module provides a connection from a 100VG network to a 10-Mbit/s Ethernet legacy network. It also has enough memory to load and run SNMP management functions for Ethernet and token ring networks.

This module slides into the hub and interfaces to the management bus and the port expansion bus on the repeater chip. The module has an i960 microprocessor and memory subsystem. The i960 is fast enough to process the counter information from the repeater chips in real time. The large 2M-byte RAM provides enough memory for both SNMP and bridging functions to run on the card.

This module also has an IEEE 802.3 controller chip to interface between 10-Mbit/s and 100-Mbit/s systems. Between this controller chip and the bus attached to the repeater

chip is an ASIC which controls the data and management information flow through the module. The module has direct access to the repeater data bus so that packet data can flow between the IEEE 802.3 controller and the 100VG network.

Acknowledgments

The introduction of the products discussed in this article was the result of tremendous effort and dedication on the part of the 100VG-AnyLAN team in Roseville and Bristol. Through the well-orchestrated efforts of the research, design, manufacturing, and marketing teams these products were successfully introduced into the marketplace, thus establishing 100VG-AnyLAN as a viable low-cost, high-speed network for the applications of today and tomorrow.

HP AccuPage: A Toolkit for High-Quality Document Scanning

Working with commercially available OCR programs, the image processing transforms used in HP AccuPage 2.0 improve the accuracy of converting scanned images from a variety of documents to editable text and pictures at the same time.

by Steven L. Webb, Steven G. Henry, Kevin S. Burke, and George Prokop

Desktop scanners are becoming more prevalent in both commercial and home office environments. Many people recognize that a scanner provides several benefits. First, desktop scanners continue to be a useful tool for improving the quality of published documents. Scanners easily allow the document author to add editable graphics to a publication, including color line drawings, diagrams, and photographs. Secondly, desktop scanners are increasingly being used to scan hardcopy documents and convert them into editable text using optical character recognition (OCR) programs. Instead of retyping text, scanner users can quickly convert documents into computer editable form, often at rates of up to 1000 words per minute.

HP continues to be a leading provider of desktop scanners for use with personal computers. HP produces color (HP ScanJet 3C) and black and white (HP ScanJet 3p) scanners that are designed to work with IBM-compatible and Macintosh personal computers. In addition to the hardware, HP also contributes software that, in partnership with software provided by independent software vendors (ISVs), provides a total solution.

HP Accupage 2.0

Two key contributions come from HP scanner software. One is to enable the scanner user to scan documents easily into a personal computer. The other is to do some sophisticated image processing on the scanned image. These two contributions help to improve the quality of the scanned image and enable much more accurate optical character recognition of text-based pages. Accurate OCR capability is provided by the image processing transforms contained in HP AccuPage 2.0.

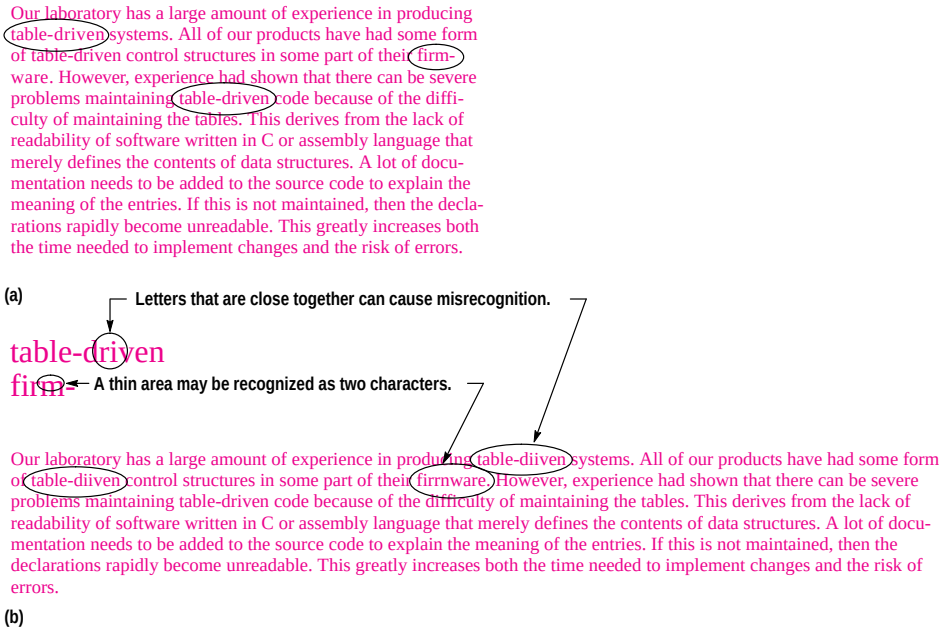


Fig. 1. A portion of a document after being read in by a typical OCR application. (a) Portion of the original document. (b) Results after scanning and passing through an OCR application.

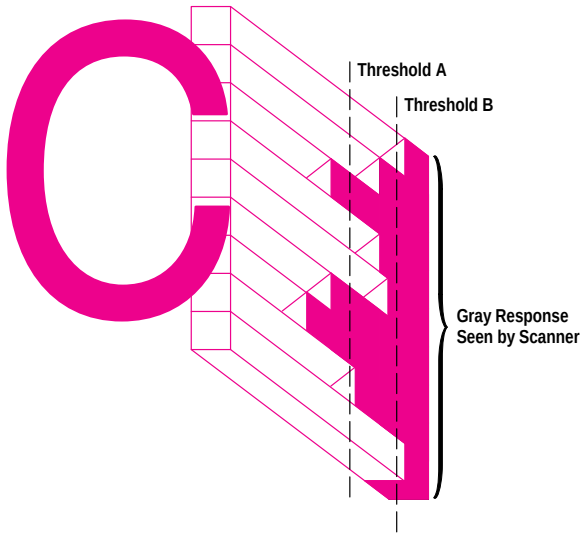


Fig. 2. A representation of the responses of the scan head as it passes over the letter c.

When HP AccuPage 2.0 is used with a third-party OCR utility, the accuracy of converting a scanned image to editable text is improved. Specifically, documents that are printed on colored backgrounds, that contain small point size text (5 to 6 point), and have complex layouts are converted with improved accuracy as a result of the HP AccuPage 2.0 image processing transforms. HP AccuPage 2.0 also allows both text and pictures to be captured from a document at the same time. This capability is becoming more important as both text and pictures increasingly make up the content of most office documents today.

In addition to providing better scanning capabilities to our customers, HP AccuPage 2.0 has also helped us to work more effectively with our ISVs to let our customers know about the value of HP AccuPage 2.0 and HP scanners. OCR independent software vendors help promote an overall document solution based around an HP scanner, HP AccuPage 2.0 technology, and the software that they provide.

This article will describe the limitations of OCR and the features provided in HP Accupage 2.0 that help to improve overall OCR accuracy. It will also describe some of the image processing techniques used to boost OCR accuracy.

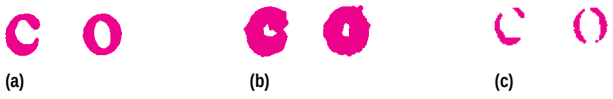


Fig. 3. The results of different threshold settings. (a) Normal setting. (b) Threshold set too low. (c) Threshold set too high.

The Limitations of OCR

The process of converting a hardcopy page into editable text accurately is not an easy task. Many documents that we can easily read with our eyes cannot be accurately converted into editable text by an ordinary OCR program. For example, Fig. 1 shows a portion of a document before and after it is scanned and run through a typical OCR program. Several of the characters weren't converted accurately despite the fact that the text is easy to read for the human eye. Current OCR pattern-recognition algorithms still require the input characters to be well-formed, smooth, and large enough so that the individual character elements are very distinct.

The reason OCR cannot recognize characters as well as people is that we have the remarkable ability to look subconsciously at different facets of what we are reading and use multilevel thinking to figure out and recognize words. We not only take into account the individual shape of each letter, but our brain is also able to piece together patterns and make use of overall context in determining the meaning of each word. Modern text recognition algorithms are only now beginning to exploit the use of context in evaluating the recognition of characters as part of words.

Another limitation of OCR has more to do with our expectations than with the algorithms. Many of us have the expectation that if we can easily read a page, then a computer-based OCR program should be able to read the page even better (more accurately and faster). In fact, we often have the hope

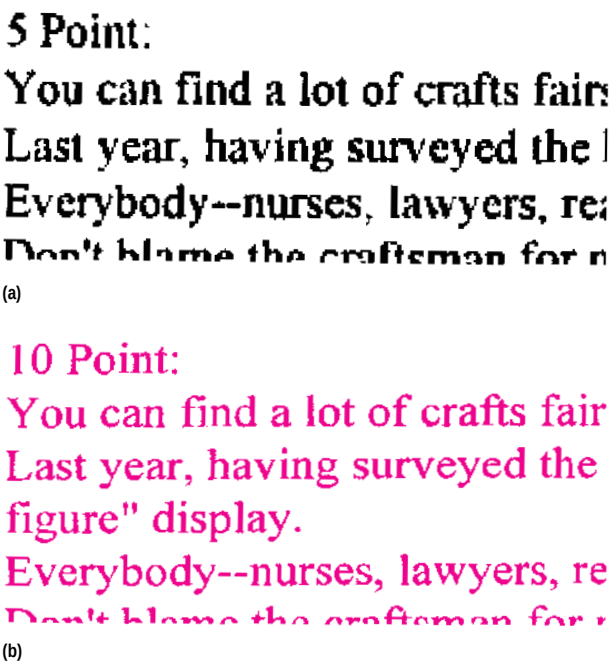


Fig. 4. A comparison between five point and ten point text scanned at 300 dpi. The five point sample has been enlarged for comparison with the ten point sample. Both of these samples are larger than actual size. Note the loss of detail in the five point sample.

Glossary

Charge-coupled Device (CCD). A CCD is a miniature photometer that measures incident light and converts the measured value to an analog voltage. The CCDs in a scanner are arranged in an array.

Desktop Scanner. A desktop scanner is a device that uses a light source, a color-separation method, and a charge-coupled device (CCD) array to capture optical information about an object (e.g., photographs or documents) and transforms that information into a digital light-intensity map for computer processing (see Fig. 1). The digital data is a two-dimensional map of pixels in which each pixel holds an intensity measurement corresponding to the reflectance (for paper) or the transmittance (for transparencies) of the object at the location represented by that pixel.

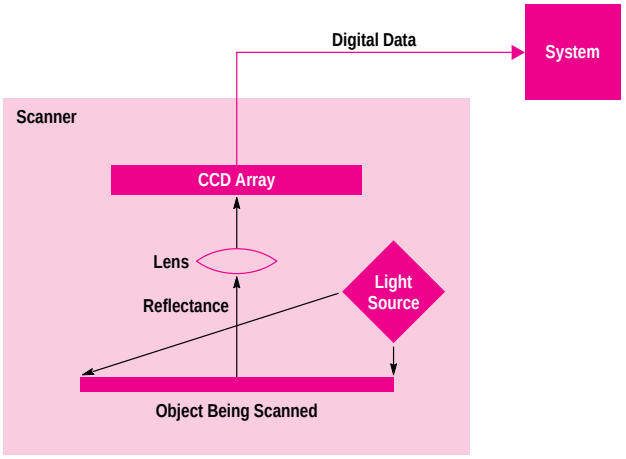


Fig. 1. The basic components of a desktop scanner.

dpi (dots per inch). The number of dots that can be printed per inch by a laser or inkjet printer.

Intensity. The amount of light reflected or transmitted by an object with black as the lowest intensity and white as the highest intensity.

Optical Sampling Rate. This is the number of samples, in pixels per inch (ppi), that are taken by a scanner per linear distance as determined by the CCD array, the optical system, and the motion of the carriage. The optical sampling rate for a scanner is specified as the pixels per inch in the x direction (across the page) and the y direction (down the page).

In the x direction, the optical sampling rate depends on the CCD layout and the magnification of the optical system. For example, a CCD with 2,550 elements applied across an 8.5-in image width has an optical sampling rate of 2250/8.5 = 300 ppi in the x direction.

In the y direction, the optical sampling rate depends on the distance and speed at which the carriage moves relative to the exposure time of the of the CCD. For example, if the carriage moves 1/300 in during a CCD exposure time, the y-direction optical sampling rate is 300 ppi.

ppi (pixels per inch). Ppi is often used interchangeably with dpi, although a dot is a bilevel entity, either on or off, and a pixel can hold multiple levels of information. For example, for an eight-bit scanner, one pixel has 256 possible values.

Resolution. For a scanner, resolution is the degree to which the scanner can distinguish detail. Resolution is dependent on items such as optical sampling rate, lens quality, filter quality, and carriage motion.

Threshold. A value to which a signal is compared when transforming from a multilevel value to a binary value. In a binary scan, parts of the image below the threshold will be recorded as black and parts above the threshold will be recorded as white.

that if the page is somewhat illegible then the computer could convert it into legible format without any errors. Thus, many people (until their expectations are set correctly) are somewhat disappointed in using today's OCR technology. However, many customers quickly realize that for many high-quality pages, the number of errors is fairly small (typically 4 to 10 errors per 2000 characters), which is favorable compared to many professional typists and very favorable compared to the ability most of us have to type text accurately.

Besides the problems that can occur with degraded characters, text recognition can also be hampered by the scanning process. For instance, like a photocopy machine, a scanner can be set to produce images that can be lightened or darkened. If the lighten or darken setting (i.e., scanner intensity) is not set correctly the OCR might produce broken or incorrect character readings. Also, when scanning text on colored background with a black and white scanner, the colored background behind the text can actually obscure the text and render it unreadable by OCR algorithms.

To understand why these problems occur it is helpful to understand something about the OCR process. Most OCR programs use a binary (only black and white) image during the process of converting a bitmap image into text. Many of the problems of character recognition can be attributed to the intensity (lightness or darkness setting of the scanner) used during the scan when the binary bitmap image is created. The small space between strokes of a character will be seen by a scanner as a shade of gray (see Fig. 2). If the intensity (or threshold) is set low then more of the gray areas of the page will be set to black. As a result of this, if the shade of gray in the small space between strokes is mapped to black, it can cause the small gaps to close in the final binary image (see Fig. 3b). This can make a lowercase c look like a lower-case o. Setting the intensity too low is also a problem if the text is printed on a colored background. If the background shade is above the threshold then the image presented to the OCR engine is solid black.

Problems also occur if the threshold is set too high, because instead of characters joining they end up with breaks (Fig. 3c).

Another problem that can occur with some scanners is that the scan won't be completely uniform across the page. This can be caused by a nonuniform light profile across the page

PC MAGAZINE

In *Living* Color

by Daniel Grotta

PC MAGAZINE EDITORS' CHOICE
February 8, 1994
HP ScanJet IIcx

PC MAGAZINE

EDITORS' CHOICE

- **HP ScanJet IIcx**
- **Microtek ScanMaker IISP**

The HP ScanJet IIcx and Microtek ScanMaker IISP are our Editors' Choice winners. Both units had the best price/performance ratio of the 21 scanners we tested. Also, our jury deemed the ScanJet IIcx and ScanMaker IISP's color output the best of this roundup, and both scanners earned outstanding scores on our test suite.

The ScanJet IIcx (with a $1,179 list price) and the ScanMaker IISP (with a $900 street price) are also great bargains, outperforming other scanners in this roundup that cost two to four times as much.

Hewlett-Packard Co. has developed a reputation for solid engineering and construction, and the ScanJet IIcx certainly upholds this tradition. The scanner's image quality and price are better than that of its recently discontinued predecessor, the HP ScanJet IIc, (an Editors' Choice in our last scanner review); the ScanJet IIcx was also the fastest scanner we tested. It comes bundled with a feature-rich Twain driver and Aldus PhotoStyler for Windows SE.

The ScanMaker IISP, which was introduced during our testing period, caught us by surprise with its strong performance on our tests. The scanner's quality color output is balanced by excellent software—a feature-rich Twain driver, Adobe Photoshop for Windows 2.5, and a new color-calibration system—all at a low $900 street price. The ScanMaker IISP was also among the fastest scanners we tested.

In terms of their price/performance ratio, the ScanJet IIcx and ScanMaker IISP clearly occupied the top tier of scanners in this roundup. After these two units, a second and third tier of performers was discernible because of their output quality. Several of these scanners distinguished themselves in particular categories—speed, line art, gray scale, or resolution (see the individual reviews for more information).

The Envisions ENV6100, which lists for a low $799, is a good buy for the price, but we did not find it to be a better value than our Editors' Choice winners. Among the stronger performers in our roundup were the AVR 8800/CLX/Pro Image, Epson ActionScanning System, Epson ES-800C Pro, KYE Genius ColorPage-1, and UMAX UC1200SE.

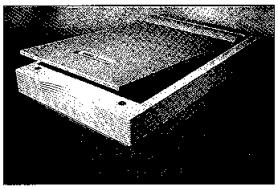
One of these color desktop scanners will likely meet the needs of most PC users who want to add photos, drawings, and text to reports, newsletters, and presentations.

Hewlett-Packard Co.

HP ScanJet IIcx

BY HOWARD ALEXANDER

Hewlett-Packard Co. has consistently come up with top-notch scanners, and the new HP ScanJet IIcx continues that trend. Set to replace the HP ScanJet IIc, the $1,179 ScanJet IIcx is a 24-bit single-pass gem that's cheaper, faster, and offers an optical resolution of 400 dots per inch, which can be interpolated to 1,600 dpi.



The price and performance of the ScanJet IIcx earned it an Editors' Choice award.

The ScanJet IIcx performs its job very well. The ScanJet IIcx was the fastest of all the units in our roundup on our speed test (28 seconds). It showed a wide color gamut with purposeful skewing toward red to enhance the brightness of images. Its color scans were the best in this review (along with the Microtek ScanMaker IISP's), and compared well to the originals.

The ScanJet IIcx's gray-scale imaging was also excellent. (For those who want a gray-scale scanner only, HP offers the 300-dpi, $879 HP ScanJet IIp.) The ScanJet IIcx showed very fine gray-scale–level sensitivity and the ability to

(continued)

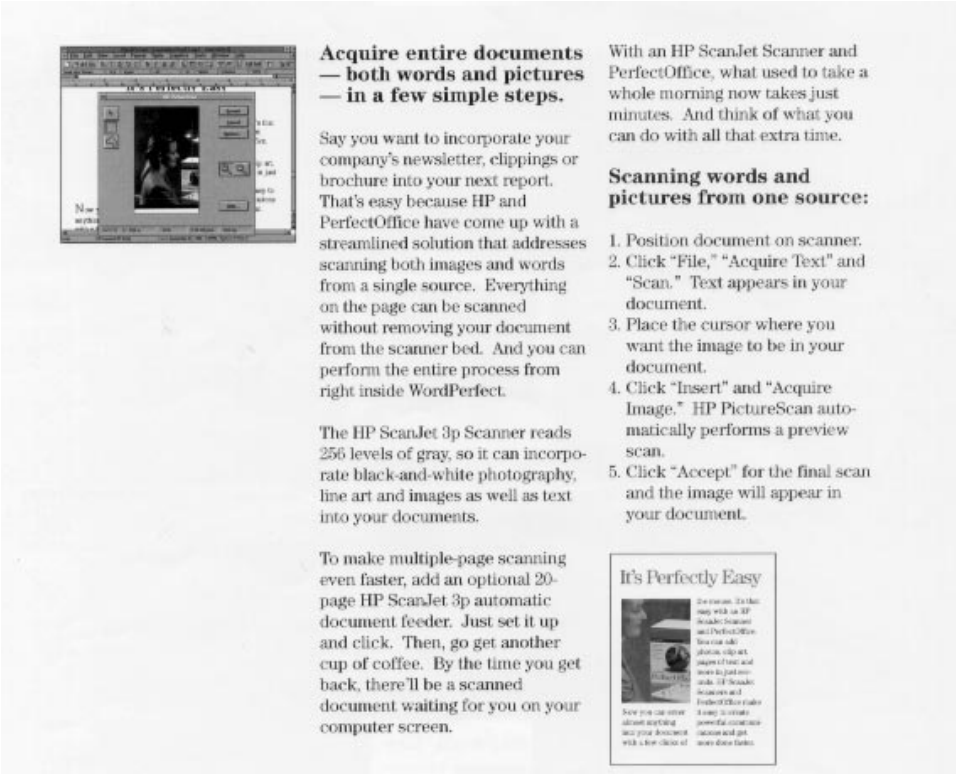
Fig. 5. Complex page with multiple columns, colors, graphics, titles, and so on. (Reprinted from *PC MAGAZINE*, February 8, 1994 Copyright© 1994 Ziff-Davis Publishing Company L.P.)

or heating effects along the length of the page during the scan. This means that the correct setting for the threshold in the top middle of the page may not be the correct threshold setting for other areas on the same page.

Another challenge for OCR algorithms is the recognition of small point-size characters (in the range of five to seven points).* Many scanners scan at a resolution of 300 dots per inch (dpi), which makes it difficult to provide well-formed character shapes to the text recognition algorithms. (See Fig. 4 for an example of five point characters versus ten point characters scanned at 300 dpi.) Small-point-size characters are frequently encountered as captions or as the fine print on legal documents. When read with current OCR algorithms at a standard 300 dpi scanning resolution, many errors result. Again, these are character point sizes that we can easily read by eye.

So far we have focused the discussion on the limitations of OCR algorithms for character and word recognition. Another important task of an OCR algorithm is to recreate the overall look or layout of the scanned page. For many typical documents, this is straightforward for existing OCR algorithms. Techniques exist to identify the location of paragraphs and to keep them in order. However, as the layout of the page gets more complicated, OCR algorithms that recognize and retain the page format begin to fall apart. Complexity can result because of elements such as multiple columns, sidebars, and graphic elements such as figures, tables, and

* There are 72 points per inch.



Acquire entire documents — both words and pictures — in a few simple steps.

Say you want to incorporate your company's newsletter, clippings or brochure into your next report. That's easy because HP and PerfectOffice have come up with a streamlined solution that addresses scanning both images and words from a single source. Everything on the page can be scanned without removing your document from the scanner bed. And you can perform the entire process from right inside WordPerfect.

The HP ScanJet 3p Scanner reads 256 levels of gray, so it can incorporate black-and-white photography, line art and images as well as text into your documents.

To make multiple-page scanning even faster, add an optional 20-page HP ScanJet 3p automatic document feeder. Just set it up and click. Then, go get another cup of coffee. By the time you get back, there'll be a scanned document waiting for you on your computer screen.

With an HP ScanJet Scanner and PerfectOffice, what used to take a whole morning now takes just minutes. And think of what you can do with all that extra time.

Scanning words and pictures from one source:

1. Position document on scanner.
2. Click "File," "Acquire Text" and "Scan." Text appears in your document.
3. Place the cursor where you want the image to be in your document.
4. Click "Insert" and "Acquire Image." HP PictureScan automatically performs a preview scan.
5. Click "Accept" for the final scan and the image will appear in your document.

It's Perfectly Easy

Fig. 6. Some of the graphics items in a document contain text, making it difficult to identify the basic object type—is it a picture, text, or something else?

photographs. Fig. 5 shows an example of a complex page that might give a typical OCR program some difficulty. Although, in most cases the errors in formatting produced by the OCR algorithms are easily identified and can be fixed, formatting errors are a major area of concern for most customers who use today's OCR products.

Page layout recognition is the job of the page segmentation algorithms. Page segmentation involves separating the page into parts that contain images of text to be converted to ASCII text and photos and drawings that are to be kept as they are. Inaccurate page segmentation causes inappropriate bitmap images to be sent to the OCR engine. Line art (or any nontextual image) sent to an OCR engine causes long delays and results in bad data being inserted into the output file. Incorrect grouping of table data can result in loss of table integrity and a loss of formatting, and incorrect column identification can cause column merging.

Once the text information has been identified and formatted, most of the current OCR algorithms completely ignore and eliminate any pictures and graphical information on the page. Many scanner users have a difficult time understanding why the picture and graphics information isn't available in the final OCR results, especially since charts, diagrams, and pictures are becoming a standard part of business communications. One challenge has been in correctly identifying the text and image regions on the page, especially when the two are located next to each other. In addition, many picture and graphical areas on a page contain textual data, which adds to the challenge (see Fig. 6). Lastly, most scanner users also expect their scanned pictures to match the original image very closely. This requires a scan to be done with moderate resolution (150 dpi) with eight bits of gray information per pixel. Most OCR software solutions today rely on 300-dpi resolution with one bit per pixel of information.

HP AccuPage 2.0 Technology

Many of the issues mentioned above have been reasonably solved with HP AccuPage 2.0 technology working in combination with the latest OCR software available from several manufacturers. When a customer purchases an HP scanner solution, they receive the scanner, HP AccuPage 2.0 software, and an OCR software package that uses the HP AccuPage 2.0 image processing transforms. Fig. 7 shows a block diagram of the components that make up the HP AccuPage 2.0 document reading solution.

By using HP AccuPage 2.0 as part of the solution, customers can get better OCR character accuracy, improved page format retention, and the ability to capture both text and high-quality images from their scanned document. Better character accuracy results from the ability to optically read text on colored backgrounds and the ability to read smaller-point-size characters. Improved page format retention and the

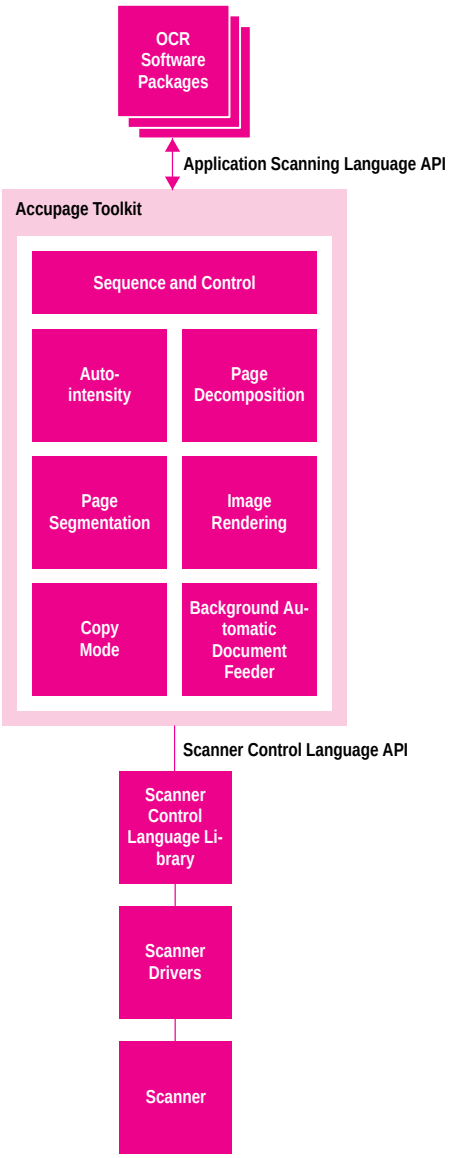


Fig. 7. The components that make up the HP AccuPage 2.0 document reading solution.

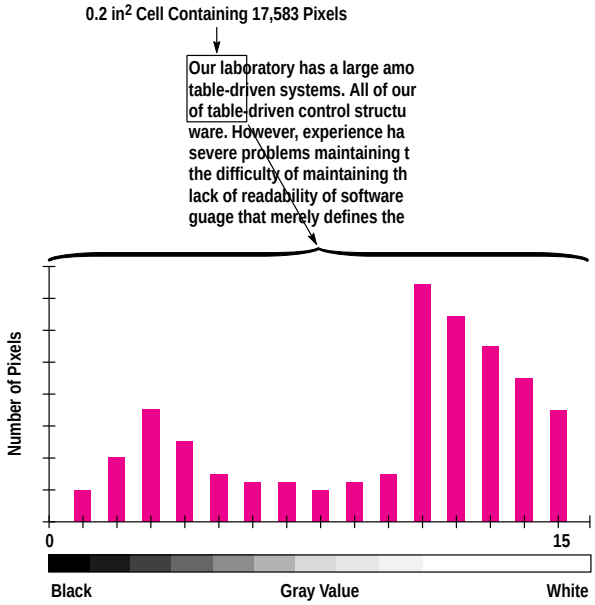
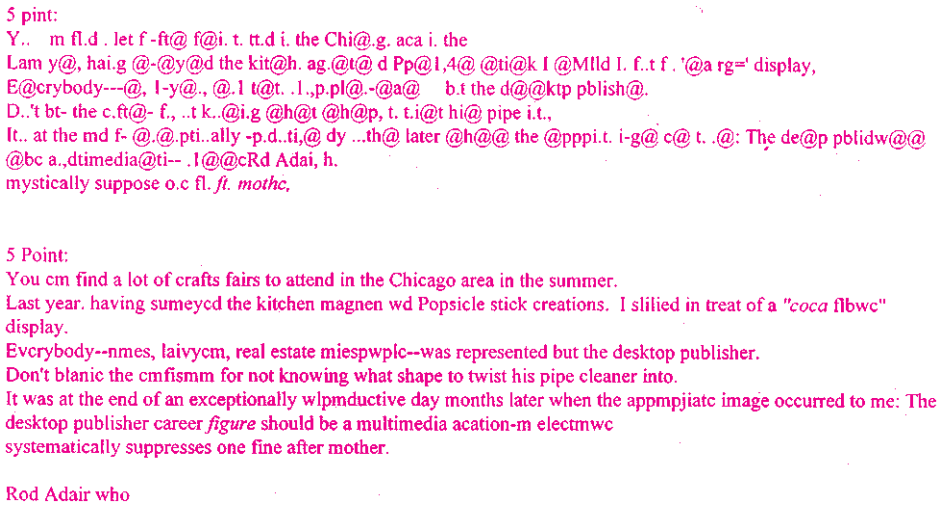


Fig. 8. Gray-level histogram. One of these histograms is made for each 0.2 in^2 area on a page.

ability to capture both text and images at the same time are the result of sophisticated page segmentation technology.

Histograms and Thresholds. HP AccuPage 2.0 deals with different color backgrounds with an adaptive threshold technique, which automatically determines the threshold level at different locations on the page. Instead of scanning the page at a binary setting, HP AccuPage scans using 16 (2^4) levels of gray. This four-bit representation of gray is mapped between 4% reflectance and 74% reflectance (the typical reflectance of a piece of paper) using all 16 levels for valid information. HP Accupage sets a threshold level for each 0.2 in^2 area on the page. The scanned image, along with an optimized threshold for each 0.2 in^2 part of the page, is sent to the OCR engine.

The HP AccuPage 2.0 optimized threshold algorithm samples the page by extracting a histogram of the pixel gray levels. This method sums the number of pixels found at each gray


5 pint:
Y.. m fl.d . let f -ft@ f@i. t. tt.d i. the Chi@.g. aca i. the
Lam y@, hai.g @-@y@d the kit@h. ag.@t@ d Pp@1,4@ @ti@k I @Mlld I. f..t f . '@a rg=' display,
E@crybody---@, l-y@., @.l t@t. .l.,p.pl@.-@a@ b.t the d@@ktp pblish@.
D..'t bt- the c.ft@- f., ..t k..@i.g @h@t @h@p, t. t.i@t hi@ pipe i.t.,
It.. at the md f- @.@.pti..ally -p.d..ti,@ dy ...th@ later @h@@ the @pppi.t. i-g@ c@ t. .@: The de@p pblidw@@
@bc a.,dtimedia@ti-- .l@@cRd Adai, h.
mystically suppose o.c fl. *ft. mothc,*

5 Point:
You cm find a lot of crafts fairs to attend in the Chicago area in the summer.
Last year. having sumeycd the kitchen magnen wd Popsicle stick creations. I slilied in treat of a "*coca* flbwc" display.
Evcrybody--nmes, laivycm, real estate miespwplc--was represented but the desktop publisher.
Don't blanic the cmfismm for not knowing what shape to twist his pipe cleaner into.
It was at the end of an exceptionally wlpmductive day months later when the appmpjiatc image occurred to me: The desktop publisher career *figure* should be a multimedia acation-m electmwc
systematically suppresses one fine after mother.

Rod Adair who

Fig. 9. Results of reading small text (a) with the small text algorithm turned off and (b) with the algorithm turned on.

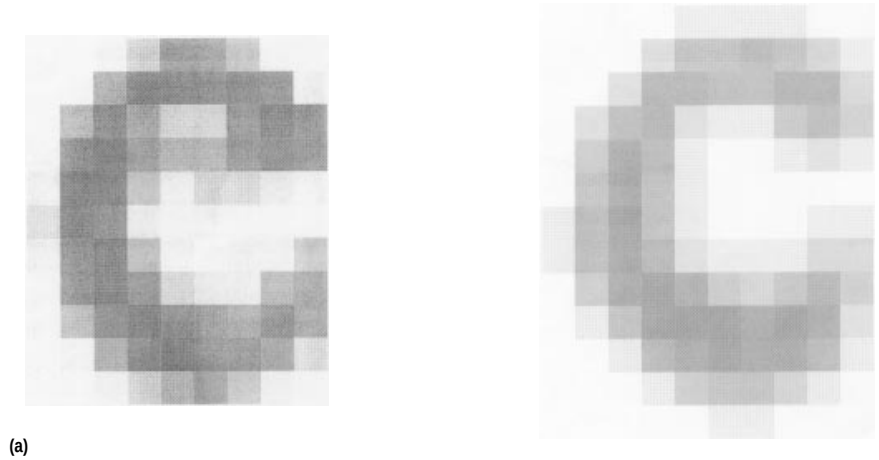


Fig. 10. Examples of doing upsampling with one bit per pixel. (a) If the threshold is too low an e starts to look like a c. (b) If the threshold is too high, an o turns into a pair of parentheses.

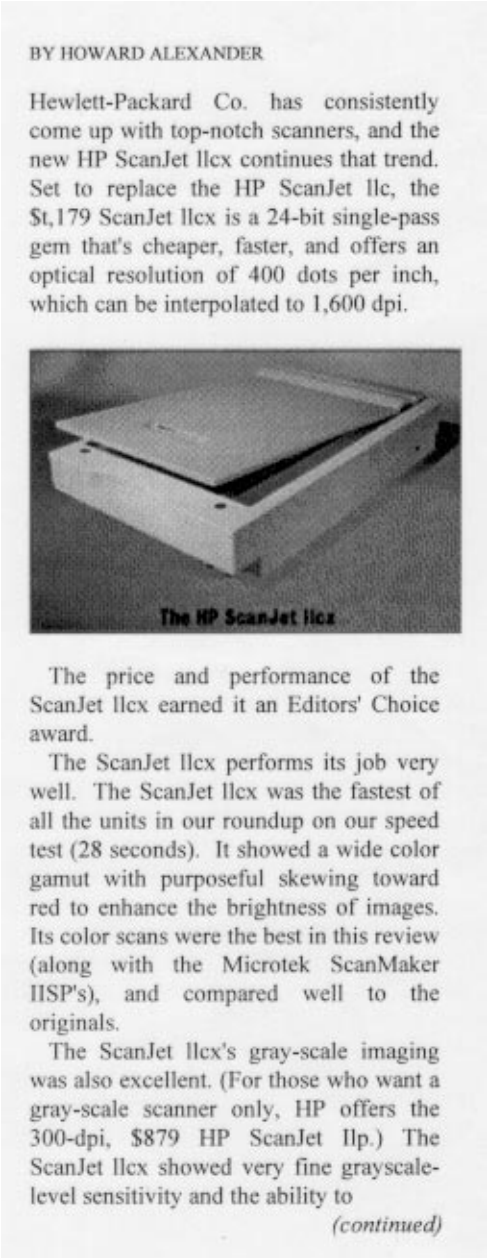
level. For four-bit data this results in a list of 16 gray levels that each contain the number of pixels found for a particular gray level. For pages that contain black text on white background the histogram will exhibit a strong bimodal distribution (see Fig. 8). The black text pixels will occupy the lower gray levels and the white background pixels will occupy the upper gray levels. Pages that contain text on colored backgrounds will also have a bimodal distribution, but not as well segmented because the background pixels will be some intermediate gray value instead of white.

The histogram allows HP AccuPage 2.0 to identify the gray levels that represent the text pixels, which should be preserved, and the gray levels that represent background pixels, which should be removed. By selecting a gray value that bisects the foreground (text) and background peaks, HP AccuPage 2.0 can convert the scanned page into a binary format that preserves the text as black and removes the background by assigning it to white. This preserves the text characters while removing the background pixels.

Small Text. HP AccuPage 2.0 is also able to recover small point-size text. The algorithm in HP AccuPage takes the four-bit grayscale information at 300 dpi and performs intelligent interpolation to provide more accurate character recognition. This algorithm takes a small amount of additional processing time, but it is able to improve recognition of text point sizes between five and seven point. Fig. 9 demonstrates the results of a scan with and without the algorithm provided in HP AccuPage 2.0.

One approach to doing this scaling might be to scan in binary (1 bit per pixel) and then upsample to 600 dpi. However, since the fine details would already have been lost, the upsampling would only provide larger versions of the same distorted characters. When closely inspected, these small character details, both the strokes and the transitions, appear as intermediate gray levels (see Fig. 10). If the threshold is too low, these fine details will be converted to black binary pixels, turning an e into a c (Fig. 10a). If the threshold is too high then the fine stroke of the small text will be converted to white pixels, converting an o to look like a pair of parentheses (Fig. 10b).

Another approach would be to tune the adaptive threshold algorithm to work well with small text. This method would



BY HOWARD ALEXANDER

Hewlett-Packard Co. has consistently come up with top-notch scanners, and the new HP ScanJet IIcx continues that trend. Set to replace the HP ScanJet IIc, the $t,179 ScanJet IIcx is a 24-bit single-pass gem that's cheaper, faster, and offers an optical resolution of 400 dots per inch, which can be interpolated to 1,600 dpi.

The price and performance of the ScanJet IIcx earned it an Editors' Choice award.

The ScanJet IIcx performs its job very well. The ScanJet IIcx was the fastest of all the units in our roundup on our speed test (28 seconds). It showed a wide color gamut with purposeful skewing toward red to enhance the brightness of images. Its color scans were the best in this review (along with the Microtek ScanMaker IISP's), and compared well to the originals.

The ScanJet IIcx's gray-scale imaging was also excellent. (For those who want a gray-scale scanner only, HP offers the 300-dpi, $879 HP ScanJet IIp.) The ScanJet IIcx showed very fine grayscale-level sensitivity and the ability to

(continued)

Fig. 11. The result of using HP AccuPage 2.0 to scan in a portion of the document shown in Fig. 5.

produce well-formed small text at 300 dpi. The problem with this is that OCR applications are tuned for standard-sized text, which ranges from 8 to 14 points. As the text strays from those point sizes, the character recognition accuracy will generally deteriorate. This is especially pronounced when the point size is reduced.

The approach HP AccuPage 2.0 uses is to scan the page in grayscale at 300 dpi. The gray pixel data preserves the details so that after upsampling, the characters contain no distortions that might decrease the OCR accuracy. After scaling the page up to 600 dpi, HP AccuPage converts the grayscale value to binary, typically using its adaptive threshold algorithm.

The algorithms for decomposing the page and separating pictures and text are another key aspect of HP AccuPage 2.0. Since some of these algorithms are in the patent application process, we chose not to describe them in this article.

Conclusion

HP AccuPage 2.0 is able to capture text and high-quality images from the scanned page. Many different image processing transforms are required to identify the image areas and retain their quality based on four-bit data input. Fig. 11 shows the capabilities of the algorithm to reproduce a page consisting of text and picture information. This capability is useful not only for OCR applications but also for document management and convenience copy applications in which the best possible rendering of the page for binary output is desired. These techniques ultimately make scanning more valuable for our customers.

Acknowledgments

We would like to thank Sue Arment, Ken Crow, Mimi Dong, Lynn Formanek, Bob Gann, Hal Hansen, Jose Moncivais, Tom Schoenleber, Christine Torrez, and Oscar Zuniga for their help with the HP AccuPage 2.0 product.

An 11.8-in Flat Panel Display Monitor

The HP S1010A flat panel display is designed to be a plug-compatible replacement for CRTs used with HP workstations. This compatibility is provided by an interface board that uses the same analog signals that drive the CRTs to create digital signals to drive a high-resolution, high-performance LCD color display.

by David J. Hodge, Bradly J. Foster, Steven J. Kommrusch, and Tom J. Searby

The HP S1010A is a color flat panel liquid crystal display (LCD) monitor. It is designed to be a plug-compatible replacement for conventional CRT monitors on HP's workstation platforms in applications requiring lighter weight, lower power, or a smaller footprint than CRT displays. The HP S1010A uses an 11.8-in diagonal active matrix TFT (thin-film transistor) color LCD module that has a resolution of 1024 by 768 pixels (see Fig. 1).

To maintain compatibility with CRTs, the HP S1010A monitor is designed to accept analog red, green, and blue input signals with composite sync on the green line* and the same video display timing used for the CRTs on standard HP workstations. During implementation we found that the available LCD flat panel displays all had digital inputs and required a clock signal. Since CRT monitors don't use a clock, the pixel clock for the flat panel display had to be generated internally.

* The red and blue signals do not have sync pulses.

Another feature of the HP S1010A monitor is its backlight saver and replaceable backlight assembly. Since their transmissivities are relatively low, color LCDs require a very bright backlight. Small, bright backlights tend to have relatively short operating lifetimes compared to CRTs. To minimize the problems associated with these short lifetimes we chose a panel with a replaceable backlight assembly. To minimize the need for replacement, we designed a circuit that can use the screen saver feature of windowing software to detect when the display is not in use and extinguish the backlight. The screen saver should be set up to display an all-black image when there is no user input for some period of time. To avoid cycling the backlight unnecessarily (for example, when the user taps the mouse as soon as the

Fig. 1. The HP S1010A flat panel display monitor weighs 12 lb including the stand, and measures 13.0 in (330 mm) wide, 14.8 in (377 mm) high, and 6.4 in (163 mm) deep.

screen saver turns on), the backlight only turns off after the black image has been present for about 15 seconds.

We incorporated a number of diagnostic features in the HP S1010A monitor. If no video is present, or the timing is incorrect and the display cannot lock onto the video signal, the display shows six horizontal color bars. Many computer system problems are hard to distinguish from monitor problems, since in either case, there is nothing on the display. The presence of the color bars tells users or service personnel that the display is functional, but that there is most likely a problem with the input video from the workstation.

The HP S1010A monitor has three LEDs that are visible from the rear of the display. One indicates whether power is present, another indicates that the backlight is turned on, and the third indicates that the display is properly locked on the input video signal. The LEDs can be used to determine quickly whether there is a problem with the power supply, the backlight, or the video.

Maintaining Plug Compatibility

Providing the capability to plug the HP S1010A into a typical HP workstation without any modification to the workstation meant that we had to design the circuits in the HP S1010A's display interface board to handle the same analog RGB composite signals sent to a CRT display. Essentially, we had to design circuits that are able to take analog signals and their associated timing and convert them to digital signals for driving counters and cells in an LCD matrix.

A conventional CRT builds an image by sweeping one or more modulated electron beams across the phosphors on the face of the tube from left to right and top to bottom. The signals that control the sweep of the beam are the horizontal and vertical sync signals, HSYNC and VSYNC respectively (see Fig. 2). As the beam sweeps across the face of the tube, it creates the image for a single horizontal line. At the end of the line, the beam is blanked, so that the retrace won't be visible, and the HSYNC signal causes the beam to sweep back to the start of the next line. The front porch provides some time for the beam to shut off before the HSYNC signal causes the retrace to begin. The back porch provides some time for the sweep circuits to stabilize before the beam is unblanked

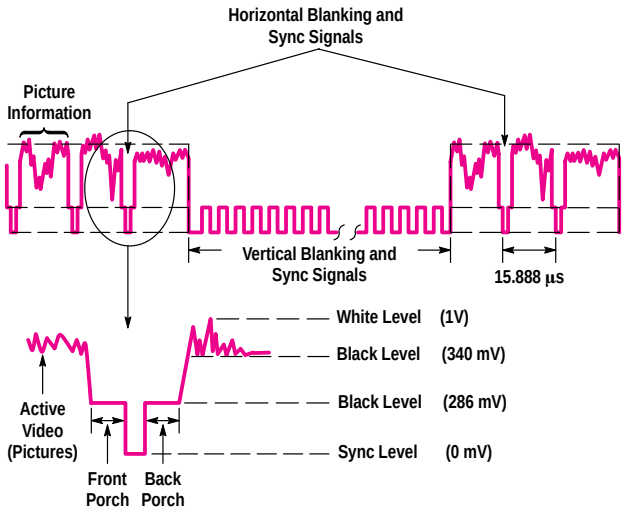


Fig. 2. Composite analog video signals.

to start the next line. At the bottom of the frame the same thing happens during vertical retrace to get the beam back to the top.

Table I shows the timing for the composite video signals coming from an HP workstation and going to the HP S1010A. To keep everything properly synchronized, the vertical timing is all in exact integer multiples of the horizontal line time for the noninterlaced monitors used in workstations. Horizontal timing is in pixel periods, which is the time required to set up and display one pixel.

Table I
Timing for the Composite Video Signals Input to the HP S1010A

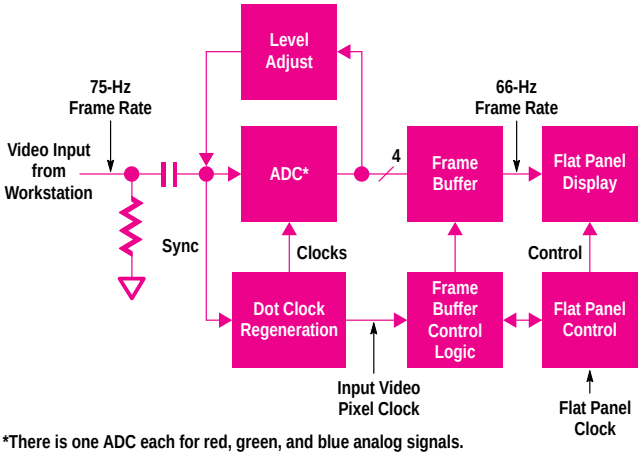
	Horizontal (Pixels)	Vertical (Lines)
Period (Active + Blanking)	1344	840
Active	1024	768
Blanking (Porches + Sync)	320	72
Front Porch	64	4
Sync	128	4
Back Porch	128	64

If we think of an active matrix LCD as a big RAM, the display update process consists of writing to all of the cells. If we always update the cells the same way, all of the addressing functions can be done by horizontal and vertical address registers inside the display. Most LCDs accept multiple pixels on each clock to keep the clock rates down. Some actually update different parts of the display at the same time. The HP S1010A's LCD takes two consecutive pixels on each clock as it scans across the horizontal lines and down the frame. After each clock, the internal horizontal address counter increments to point to the next pair of pixels. At the end of the line, there are two things that need to happen: the horizontal counter needs to be reset and the vertical counter needs to be incremented. The signal that performs this function is very similar to the horizontal sync signal in a CRT, so we call it HSYNC. After the last line at the bottom of the frame, we need to reset the vertical address counter to start over at the top of the next frame. Since the signal to do this behaves very much like the vertical sync in a CRT, we call it VSYNC.

One important thing to note about the timing of CRTs and LCDs is that the LCD has much more flexibility. The CRT sweep circuits must drive the large inductive load of the deflection coils, so it is not practical to adjust the timing dynamically. Within limits, the LCD's timing can be varied. We took advantage of this characteristic in the design of the HP Sl0l0A.

Hardware Architecture

Fig. 3 shows a block diagram of the main components of the HP S1010A flat panel display monitor. The analog video from the workstation comes into the termination network which matches the impedance of the video cable. The video then goes to the analog-to-digital converters (ADCs). The digital output is sampled and used to control the level adjust



*There is one ADC each for red, green, and blue analog signals.

Fig. 3. A block diagram of the components in the interface board for the HP S1010A flat panel display monitor.

circuits which set the dc level of the ADC inputs and the ADC reference voltage so that the entire dynamic range of the ADCs is used. The video sync signal is extracted from the input video and sent to the dot clock regeneration circuit which generates the clock signals used by the ADCs to sample the video signal. The digital outputs of the ADCs are sent to the frame buffer which synchronizes the input signal to the flat panel timing and sends it to the panel.

Analog-to-Digital Conversion

The ADC circuitry converts the incoming video analog stream into the digital data that the flat panel display can accept. In the case of the HP S1010A monitor, 84 Mpixels/s are converted so that the four most-significant bits of each color's digital representation do not vary from one frame to the next because of digital noise. One of these four bits is used for control, and the other three bits define the three-bit color that the flat panel uses to generate eight intensity levels for red, green, and blue, providing a total of 512 displayable colors. The one-volt swing of the input analog video is divided into three main regions: the sync level, the blank level (typically 286 mV above the sync voltage), and the video level (ranges from 54 mV above blank signifying black to 714 mV above blank signifying white). These voltage levels are shown in Fig. 2. The period of the sync signal on the analog input is the time required to provide data for one horizontal line of pixels. On the HP S1010A monitor this time is 15.888 μs.

Fig. 4 shows that the ADC circuitry is cleanly divided between the dot clock regeneration circuit and the support circuitry for the ADC. The dot clock regeneration circuit uses the horizontal sync signal from the green analog input line to create a clock which is used by the ADC to sample the analog video signal. The ADC support circuit is primarily concerned with controlling the offset of the input analog signal and the reference voltage to ensure that the ADC outputs have the desired digital range.

Liquid Crystal Display Technology

Liquid crystal displays, or LCDs, are divided into two main classes: active and passive matrix LCDs. Passive matrix displays scan each of the cells, or pixels, sequentially. They are less complex and less expensive than active matrix devices, but the addressing technique they use means that each cell is only driven for a small fraction of the time. To maintain the image, the cells must hold their state for a long time (analogous to long decay phosphors in a slow-scan CRT). The disadvantage is that the response time of the display is slowed, which leads to ghosting on rapidly changing images, such as when the cursor is moved.

Active matrix displays have circuitry associated with each cell. The usual technique for building active matrix LCD circuits is to use a thin film of silicon grown on the display glass. This technique is known as thin-film transistor, or TFT. This type of LCD can be thought of as a big dynamic RAM, with one cell for each pixel. The RAM drives each liquid crystal cell continuously, and the RAM cells are refreshed by scanning the display. This enables the liquid crystal cell to be faster, improving response time dramatically. It also allows the display to have much higher resolution, since the drive time of each cell is not reduced by adding more cells. The increased resolution makes it more practical to build a color display since a color display has at least three times as many effective pixels as a monochrome display (one red, one green, and one blue subpixel for each pixel).

LCDs offer a number of advantages over CRTs. Because they are fabricated with a lithographic process, they offer excellent linearity, convergence, and purity. LCDs have no electron beam, making them unsusceptible to magnetic fields. They have lower power requirements (about 55W versus 100W for a comparable-size CRT). Because conventional CRTs need to deflect an electron beam, they must have a greater depth, and thus a larger footprint, than an LCD monitor. CRTs need to accelerate the electron beam, which requires high voltages that are not necessary for LCD monitors. Side effects of the high voltages include x-ray emissions and potential electrostatic problems in some environments. Finally, LCD monitors don't need a heavy glass bottle to maintain a vacuum, so they weigh much less than CRTs.

Accurate Positioning

To create a pleasing visual image without pixels jittering around, it is critical to position the sampling clock edges precisely and repeatably. If a pixel is sampled at horizontal position x in one frame, but at position x+1 in the next frame, the user will easily see the difference as noise on the monitor. Furthermore, the sample must be taken when the signal is stable and not transitioning between pixels. This requires the sampling edge precision to be significantly less than a single pixel time. The HP S1010A is able to recover the digital pixel data very cleanly.

For the HP S1010A monitor, positioning the sampling edge requires that it regenerate a dot clock with exactly the same frequency as the clock used in the workstation to generate the analog video data. The phase-locked loop circuit is used to synchronize an internally generated sync pulse with the horizontal sync pulse on the green video line. The horizontal sync signal provides only one synchronization event every 15.888 μs. At 84 Mpixels/s, pixel time is only 11.82 ns, allowing 1344 pixels (one horizontal line of data) between each synchronization event. The transition time on the analog input from one level to the next is 4 ns worst case, leaving only 7.82 ns for setup and uncertainties over all temperature, voltage, and component variations.

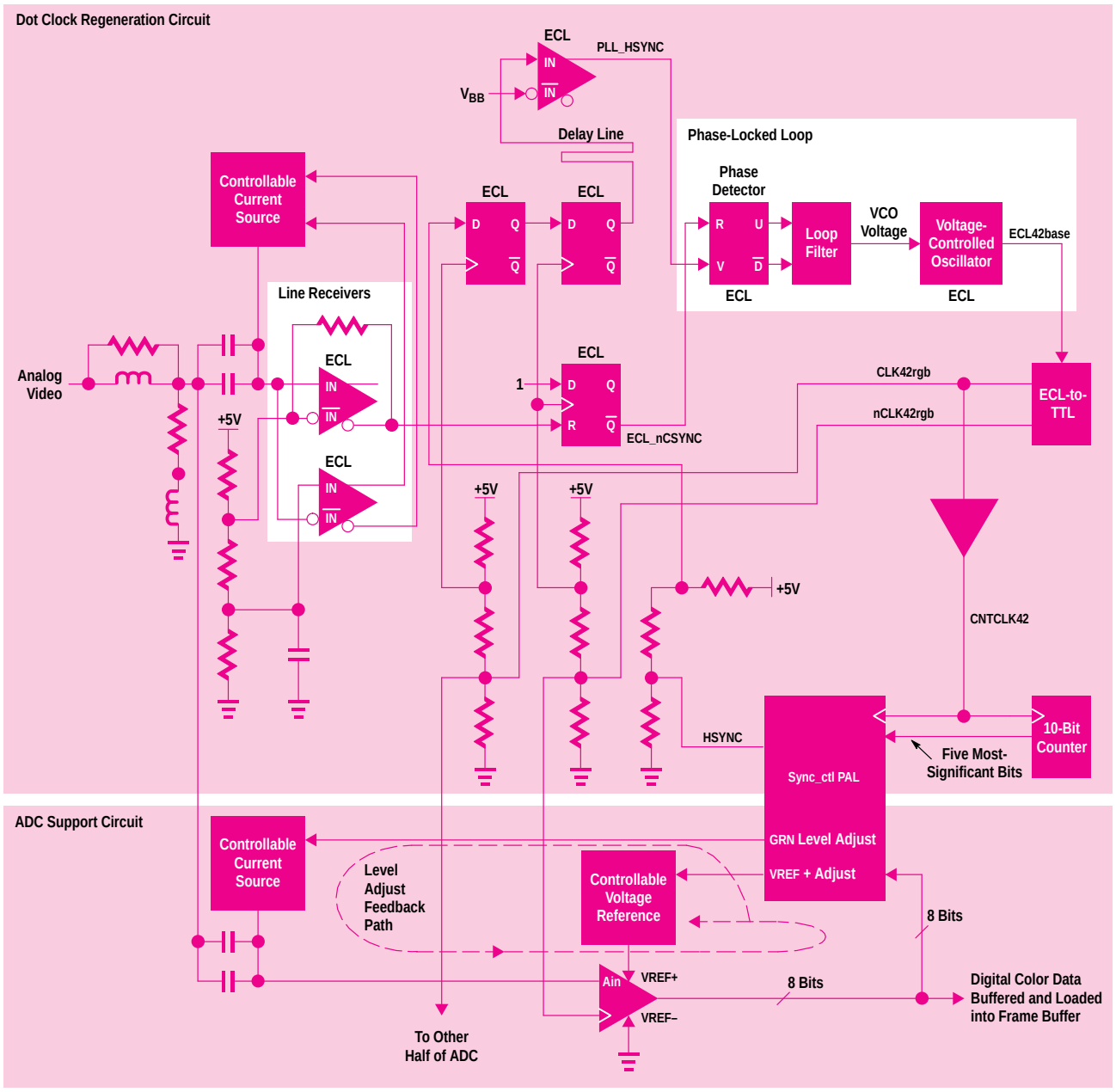


Fig. 4. The analog-to-digital conversion circuitry in the HP S1010A interface board.

Skew can be introduced in the three major areas: sync separation (extracting the sync signal from the analog input to be used by the phase-locked loop), internal clock skew (skew control between the edge of the internal sync signal and the clock sent to the ADC for sampling the data), and phase locking (degree to which the internal sync signal can be matched with the analog input sync signal).

Fig. 5 shows how the signals used in the HP S1010A monitor line up. Ultimately, we are trying to get the rising edges of CLK42rgb and nCLK42rgb to be precisely positioned with respect to the analog input. This is because these signals are used to clock the ADC that produces the digital pixel data.

The ECL_nCSYNC signal is generated from the analog video signal after the analog video signal has been terminated and voltage shifted through bypass capacitors. Separating the sync signal from the analog video signal is a task that all display monitors must do and there are many parts commercially available for this purpose. However, none of these parts were acceptable for use in the HP S1010A monitor because of the great uncertainty about the propagation delay of these parts. Therefore, we used a carefully biased ECL differential line receiver (MC10E416) arranged in a Schmitt-trigger configuration. The MC10E416 is sensitive enough to distinguish between the sync voltage on the video input and the blank voltage which is only 286 mV higher. The MC10E416 has a small propagation uncertainty; its speed ranges between 200 ps and 550 ps, giving a net uncertainty of only 350 ps in propagation delay. Another differential line receiver is used to control the offset of the analog input to the first receiver. This biases the sync-to-blank transition voltage into the range where the MC10E416 is most sensitive.

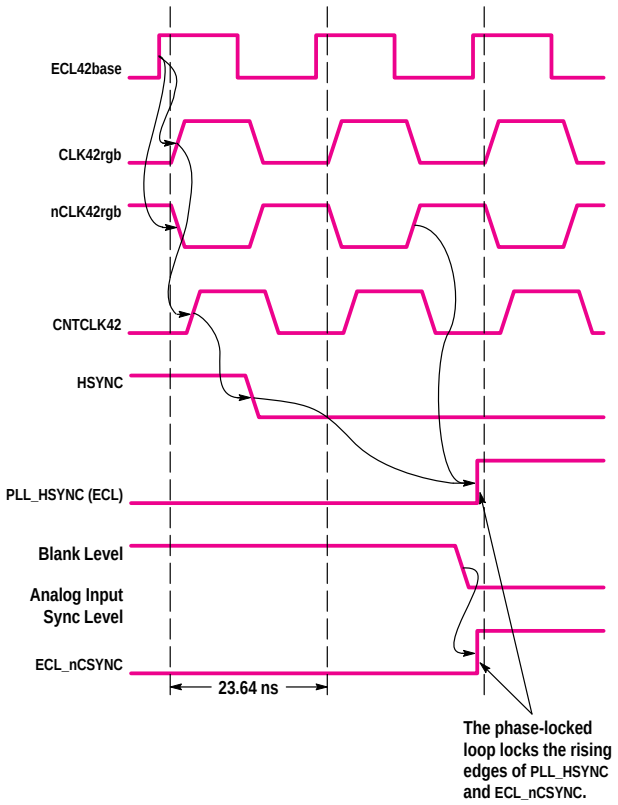
Generating the internal sync signal (PLL_HSYNC) begins with the signal ECL42base, the output of the voltage-controlled oscillator in the phase-locked loop circuit. ECL42base is converted into TTL (CLK42rgb) and then buffered to the rest of the system as CNTCLK42. A set of counters with the appropriate control logic generates the TTL HSYNC signal which has a period equal to 1344 84-MHz clock periods (one full input line time). This signal is delayed by one and a half 42-MHz clock periods to produce one input to the phase detector in the phase-locked loop circuit. The final flip-flop, which generates PLL_HSYNC, is clocked by a simple TTL-to-ECL resistor ladder from nCLK42rgb, one of the clocks used for the ADC circuit.

Because of its high speed, ECL technology was used in time-critical functions in the dot clock regeneration circuitry, and TTL was used elsewhere because of its low cost.

After being generated by the flip flop, PLL_HSYNC goes through a delay line and then through one of the line receivers on the same MC10E416 that does the sync separation for the signal ECL_nCSYNC. The delay line compensates for all propagation delay shifts and results in correctly positioning the CLK42rgb positive edge within the analog video pixel period. The pass through the line receiver is done primarily to minimize skew uncertainty contributed by the line receiver itself. Passing HSYNC through the same part eliminates the 350-ps skew uncertainty mentioned above and allows us to use the specification for "within-device skew" which is only 50 ps! The fact that the MC10E416 also buffers the delay line from the sensitive phase detector of the phase-locked loop is an added bonus. Similarly, ECL_nCSYNC passes asynchronously through the same part used to generate PLL_HSYNC so as to minimize the same uncertainties mentioned above.

Throughout the synchronization process, the goal is to minimize the skew (actual propagation times are irrelevant). The commercial sync separators with propagation delays varying from 5 ns to 25 ns were unacceptable, but a part with a minimum delay of 24 ns and a maximum delay of 25 ns would have been acceptable, since another clock cycle and a delay line could have been accommodated. The end result is a 6-sigma skew budget of 11.44 ns, which is barely within the 11.82 ns pixel time.

One final issue in dot clock regeneration is the behavior of the phase-locked loop which locks the edges of PLL_HSYNC and ECL_nCSYNC by modifying the frequency of ECL42base. The phase detector controls the VCO voltage such that a 1-ns difference at one HSYNC edge will cause almost a 1-ns phase shift by the next HSYNC edge. However, because of the loop filter used, the average period of HSYNC changes little. If the HSYNC period is correct but out of phase by a few nanoseconds, on the next cycle the HSYNC period will still be correct, and the phase will no longer be shifted. Mathematically, this creates a response for the entire phase-locked loop system that is almost exactly critically damped. Lock is achieved quickly (less than 2 ms after connection to the analog input signal) and HSYNC drift is minimized. The phase detector we chose is one of the best on the market and can detect edge differences between PLL_HSYNC and ECL_nCSYNC of as little as 300 ps, which is referred to in phase-locked loop literature as the "dead zone."



Signal	Source or Purpose
ECL42base	Output of the Voltage-Controlled Oscillator
CLK42rgb	TTL Clock Used by the ADC to Sample Data
nCLK42rgb	Inverse of CLK42rgb also Used to Sample Data
CNTCLK42	Clock Used to Count Pixels
HSYNC (TTL Version)	Regenerated Horizontal Sync from Pixel Counts
PLL_HSYNC nCLK42rgb	ECL Version of HSYNC, Precisely Linked with for Locking onto the Input Sync
ECL_nCSYNC	Input Composite Sync Used for Locking to Internally Generated HSYNC

Fig. 5. The timing of the phase-locked loop input signals.

Automatic Scaling

To make the best use of the eight shades per color provided by the flat panel display, the HP S1010A uses feedback to control the ADC so that the full digital range is available. The control logic uses timing information from the dot clock counters to determine when the video is exhibiting a blank voltage and when it is exhibiting a sync voltage. The GRN (green) level adjust signal in the sync_ctl PAL in Fig. 4 is used to raise or lower the analog voltage to set the sync level at a digitized pixel value of 74.5. If the digitized data collected during the sync period is 74 or less, the GRN level adjust causes the input to the ADC to rise, whereas if the digital data during sync is 75 or more, it causes the ADC input to go down. In a similar fashion, VREF adjust is used to set the blank level to an eight-bit value of 123.5.

Setting sync (0 mV on the analog input signal) to 74.5 and blank (286 mV) to 123.5 gives a full white (1000 mV) level of 246 (Fig. 2). The most-significant bit of the ADC output is low during sync and blank and high during active video. The next four bits from the ADC represent the pixel value sent to the frame buffer. Since the flat panel uses only three bits, any ADC output between 240 and 255 will cause the

pixel to be displayed at full brightness. If the display is connected to another system with an analog output that is consistently 5% higher, sync will still be at 74.5 (still a 0-mV input), blank will be adjusted to 123.5 (now a 300-mV input), and full white will still be 246 even though the analog signal is at 1050 mV.

With a consistent digital translation being made on the voltage waveform, the input can be optimized for noise immunity. Typically, the input analog video will have 256 shades for each red, green, and blue signal, and with 660 mV for video, this is less than 3 mV per color. If all colors are allowed on the input, 3 mV of noise will be noticeable in the color sampling. On a CRT monitor, 3 mV of noise from frame to frame will be unnoticed by the user since this is such a small intensity difference. However, on the HP S1010A monitor only eight shades of each color are available, so any color change will be noticed. If noise appears on a color that lies near the digital sampling transition from one of the eight color levels to the next, this 3 mV of noise may cause the color to be 1/8 brighter or dimmer from frame to frame, which will clearly be noticed by the user. Our solution to this problem is called color centering and involves a software modification to host systems capable of creating an analog input to the HP S1010A monitor. This modification simply maps the 256 normal output shades into one of the eight levels that the HP S1010A is able to recognize with large noise margins. The host system performs this mapping by controlling what is written into the color map of the output RAMDAC (random access memory digital-to-analog converter) that generates the video.

Frame Rate Matching

One obstacle we had to overcome in the design of the display interface board was matching the slower frame rate of the LCD panel to the faster frame rate of HP's typical workstation video timing. To solve this problem, we included a full-frame VRAM frame buffer. Even with a frame buffer the two data streams have to be properly synchronized or the faster video input side will eventually catch up and pass the flat panel side sometime during the visible part of the frame. This can cause a tearing phenomenon in which part of the displayed frame on the flat panel comes from one input frame and the rest of the displayed frame comes from an other input frame. When the image is changing rapidly, this phenomenon is visible to the user. To avoid this artifact, we have to store the digitized video information and shift it out to the flat panel display in such a way that only whole input frames are displayed, while still satisfying all of the flat panel display requirements.

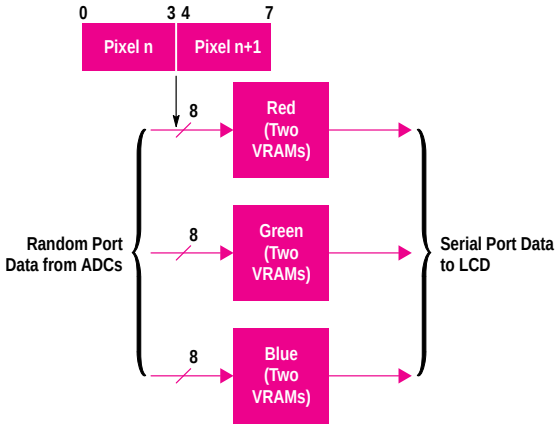


Fig. 6. The frame buffer arrangement in the display interface board for the flat panel display.

One aspect of this frame buffer is that neither data stream (digitized video into the frame buffer or display data to the flat panel display) can be interrupted. This is not usually the case in typical graphics systems because the designer can interrupt the flow of data into the frame buffer when necessary. For the HP S1010A flat panel design an algorithm had to be devised to write and read the frame buffer in such a way that neither data stream is interrupted, while keeping things synchronized to prevent video tearing.

The digitized video feeding into the frame buffer is running at a 75-Hz frame rate, but the LCD monitor is not capable of running at more than about a 66-Hz frame rate. The ratio of the two frame rates is about 9:8, so we chose to match them by discarding every ninth frame of input video. If this is done synchronously, so that only whole frames are discarded, the user will not notice the skipped frame. In fact, the only effect will be a delay of no more than 28 ms in updating the display if the image changes during the skipped frame.

VRAM Frame Buffer Architecture

A four-pixel architecture is used for the HP S1010A monitor. Thus, every frame buffer write cycle stores four pixels of information. Each pixel has three colors, with four bits for each color (only three of these bits are used for color resolution so the least-significant bit is not used). Therefore, each RAM write cycle will write 48 bits (four pixels × three colors × four bits/color) into the frame buffer.

The frame buffer consists of six 256K × 8-bit video RAMs (VRAMs), arranged in two banks of three. Each VRAM stores the data for one color (red, green, or blue) in two adjacent pixels in a single word. The digitized input video is written through the random port of the VRAMs, and the flat panel video is read from the serial port (Fig. 6).

Two things should be noted about this architecture. First, all six VRAMS are controlled by the same control signals (RAS, CAS, DSF, WB/WE, DT/OE, and so on). This simplifies the control circuitry because separate signals are not required for each bank. Second, an entire horizontal line of information (1024 pixels) can be stored in half a row in the VRAM array. This also simplifies the control circuitry because each line fits in a single half of the split serial port register.

To simplify the control logic, the frame buffer begins to store data from the ADCs at the end of a vertical sync period. Since the vertical back porch (the delay between the end of sync and the start of displayed data of the digitized video) is 64 lines, the first 32 rows of data in the VRAMs are meaningless. The counters in the serial port control block are preloaded with the correct value so that the first 64 lines of data from the frame buffer are never displayed. This scheme eliminates the extra counter that would be needed to count scan lines to get through the vertical back porch.

(continued on page 58)

Product Design of the HP S1010A Flat Panel Display

Simplicity and elegance were the two main underlying objectives for the product design of HP's first standalone flat panel display monitor. Because of the high cost and resolution of the display technology, the product design needed to radiate innovation and quality. The use of many subtle curves gave the product a very soft and sophisticated look and feel.

Other mechanical objectives were to design a small-footprint, yet stable package with a wide tilt range and swivel, require no fan, be desktop or wall mountable, and have built-in security and cable management features.

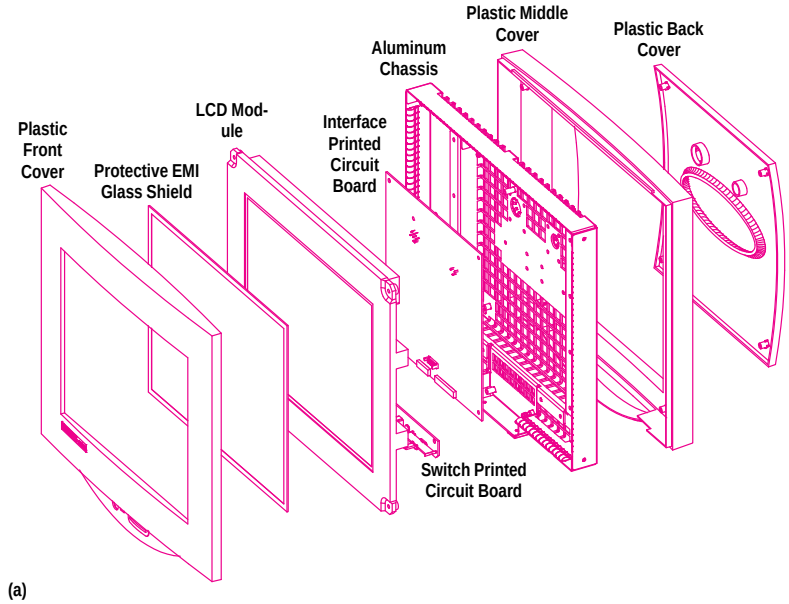
Simplicity

The design is made up of two assemblies: the chassis/display assembly which houses the display module and control electronics, and the stand assembly which provides structure and dynamic movements (see Fig. 1). The stand has no electronics and is detachable. The overall structure is C-shaped which helped to reduce the footprint by balancing the display over the stand and provided a wider tilt range (Fig. 2). It also gave it an elegant, floating display look. An added benefit to the C shape is that a keyboard can fit under the display portion to free up even more desk space.

Because of the schedule and available engineering resources, simplicity was taken seriously. A human factors study was completed giving the desired height, tilt, and swivel ranges. However, designing individual height, tilt, and swivel adjustment mechanisms would have taken more time and resources and potentially resulted in a bulkier design. For simplicity, a fixed height with a wider tilt range was decided upon that would meet most users' needs. As for swivel, the simple answer was: just slide it around. With the appropriate feet material, the monitor is light enough to be easily swiveled and slid anywhere on the desk. For thermal, size, and simplicity reasons, it was decided not to incorporate the power supply and instead use an external power module (off the desk, out of sight).

Chassis/Display Assembly

This chassis/display assembly shown in Fig. 1a consists of the LCD module, the interface printed circuit board, the power and brightness switch board, an aluminum chassis, a protective and conductive glass over the display, and cosmetic plastic covers. An aluminum chassis (as opposed to steel) was chosen to reduce weight and for EMI containment. The chassis contains a stainless-steel gasket to provide EMI contacts around the video and power connectors. A steel bracket is attached to the rear to provide a more rigid mounting location for the hinge and stand assembly. The plastic middle and back covers are heatstaked to the metal chassis. The printed circuit boards are snapped and then screwed into place. A protective glass, which is conductive and provides EMI containment, is taped to the display module metal housing. The display module is connected via cables to the control board and then screwed to the chassis. The plastic front cover hooks at the top on the middle cover and is then screwed underneath into the chassis. The



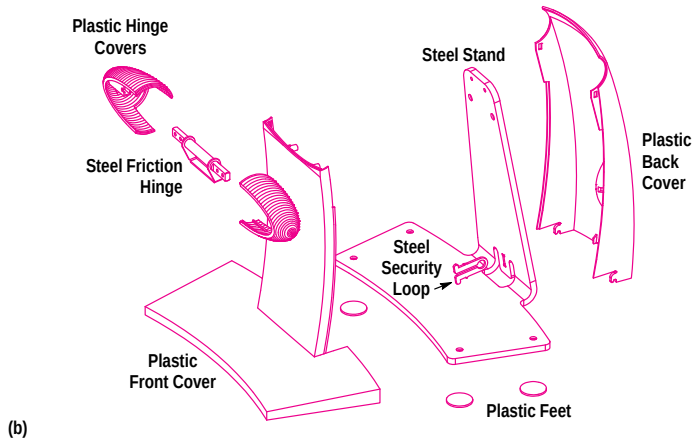


Fig. 1. The two assemblies that make up the product design for the HP S1010A flat panel display. (a) The chassis/display assembly. (b)The stand assembly.

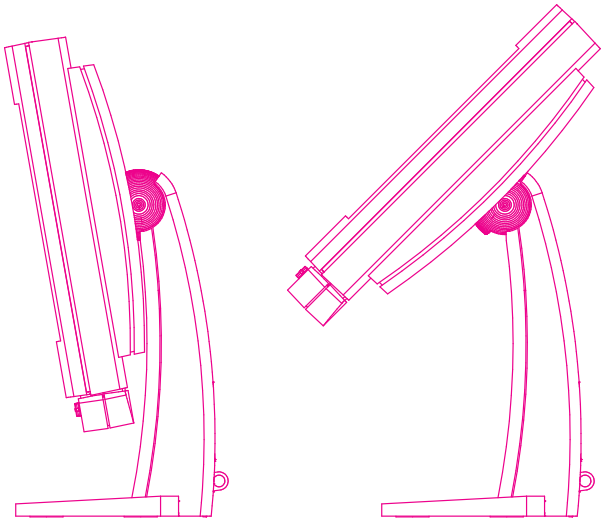


Fig. 2. The tilt range of the flat panel display.

assembly weighs approximately 6 lb (2.7 kg) with the LCD module weighing approximately 3 lb (1.4 kg).

The Stand Assembly

To provide a stable base for the display assembly, the stand was designed out of heavy sheet steel with a counterbalancing shape (Fig. 1b). The stand assembly includes a custom steel friction hinge, a stainless-steel security loop, and cosmetic plastic covers. The security loop snaps into the metal stand. The plastic front cover is heatstaked to the metal stand. The plastic hinge covers are screwed to the hinge shaft. The hinge assembly is screwed to the stand. The plastic back cover, which incorporates a cable management recess, is hooked at the bottom into the front cover and rotates and snaps at the top into the front cover.

Final Assembly

The stand assembly is mounted to the display assembly via two screws. To expose the mounting holes, the back stand cover is snapped off and the hinge shaft is aligned with the chassis mounting bracket and secured with two screws. Fig. 3 shows different views of the final assembly of the display.

Conclusion

As a testimony to our adherence to the original design goals of simplicity and elegance, the product has won two major design awards: Design Zentrum Red Dot for High Design Quality (Germany 1994) and The Industrial Design Excellence Award-Gold 1994 (United States), featured in the June 6, 1994 issue of *Business Week*.

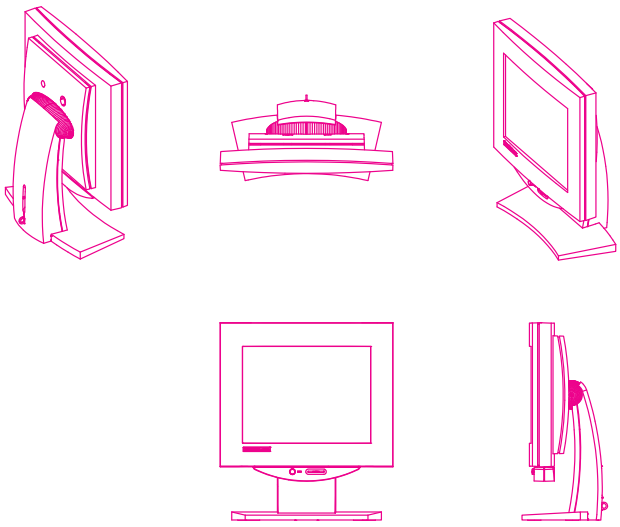


Fig. 3. Different views of the HP S1010A flat panel display.

VRAM Frame Buffer Control

The frame buffer control is responsible for, among other things, deciding when to do split data transfers from the VRAM array to the serial port shift registers and controlling which incoming frames get written into the frame buffer. (See a "A Note About VRAMs," on page 59 for some definitions of the terms used in this section.)

To ensure that the flat panel display always has the correct information to display, a split data transfer must occur on the random port side of the VRAMs for every line on the display. Table II shows the differences in the horizontal and vertical rates for the digitized video coming from the workstation and going to the LCD monitor.

Since the horizontal rate of the incoming video is faster than the horizontal rate of the LCD monitor, we are guaranteed that if we choose a point in the incoming horizontal period (say when we transition to horizontal blank) and do a single split data transfer every time we reach that point, we will always do at least one split data transfer for every horizontal line on the monitor. Although this seems to be straightforward, the scheme is complicated by a timing constraint of the VRAMs which prohibits split data transfers too close to the time when the monitor ends its current line. Fig. 7 shows the times (A and B) where split data transfers are not allowed.

Table II
Horizontal and Vertical Rates

	Horizontal		Vertical	
	Displayed µs	Total µs	Displayed ms	Total ms
Digitized Video	12.105	15.888	12.202	13.346
LCD Monitor	18.954	19.250	14.784	15.015

This requirement can be met if only one data transfer is allowed per flat panel horizontal line (one period of flat panel horizontal sync). This is achieved by looking at the sense of the qsf signal at the present time and comparing it to the sense it had the last time a data transfer was performed. If they are the same, this data is already in the shift register, so no data transfer is done. If they are different, a data transfer is done. In this way, no data transfers are performed at times A or B.

Every 15.888 µs there is a potential data transfer window. If a data transfer window falls in the middle of time B, that transfer will be inhibited because there will already have

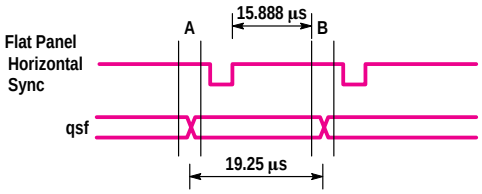


Fig. 7. Restricted split data transfer times.

been a transfer earlier in the same flat panel line. The qsf signal comes from one time domain and must be synchronized to another time domain before it is used to enable data transfers. By the time a change in qsf propagates through the synchronization and setup logic, time A is past, so no data transfer will occur there either.

Frame Rate Synchronization Algorithm

Data transfer in and out of the frame buffer must be synchronized to prevent the tearing phenomenon described above. Our synchronization technique does not try to synchronize the front-end clock (the output of the phase-locked loop) with the back-end clock (from an on-board oscillator). Instead, our technique uses events on the faster video input side to trigger events on the flat panel side. Specifically, for every eighth flat panel frame, the back end holds off asserting the vertical sync to the panel until it receives a vertical sync from the input video side. By doing this every eight frames and choosing the frequency of the back-end oscillator carefully and adding extra vertical front porch lines during the previous seven frames, a robust "on the fly" synchronization algorithm can be implemented.

Fig. 8 shows that there are only four states in the monitor's vertical state machine. The Nonsync state represents the time when the flat panel display is receiving active video (i.e., the 768 lines of digitized input video). The Sync Pulse state represents the time when the vertical sync signal is sent to the flat panel display. The other two states (Extra Lines and Holdoff) represent the times when the flat panel display is in its vertical front porch. The flat panel doesn't require any vertical front porch lines. However, to synchronize the incoming and outgoing frames in the frame buffer to avoid video tearing, we needed to add a few extra lines to the vertical front porch portion of the flat panel display to

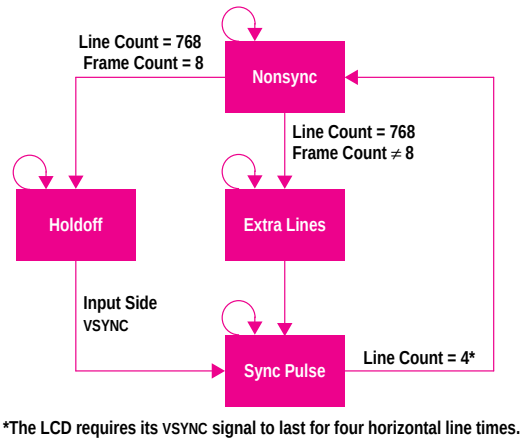


Fig. 8. The vertical state machine for the HP S1010A monitor.

A Note About VRAMs

Video RAMs, or VRAMs, are a variety of two-port dynamic RAM. They are designed to work well in graphics and video applications. The main port allows random access to any cell of the RAM. The other port consists of shift registers that are controlled by an independent clock. In the HP S1010A, the random port runs in the video input clock domain, and the serial port runs in the flat panel clock domain.

A data transfer operation loads the shift registers with data from the RAM array. The shift registers can be treated as two semi-independent halves, so that one half can be loaded without interfering with the data being shifted out of the other half. This provides more flexibility, since a data transfer operation (called a split data transfer in this case) can happen at any time while the other half is active, and transfers can be arranged so that there will be no interruption in the data flow out of the shift registers. The VRAM provides a signal called qsf to indicate which half of the shift register is active. When the data in the active half of the shift register is exhausted, qsf toggles, and the other half becomes active. This signals the HP S1010A's control logic that it's time to get ready for another split data transfer.

get an even ratio of eight flat panel frames for every nine input video frames.

After transfer of the last active line of video to the flat panel display, the state machine goes to the Extra Lines state where it will stay for eight horizontal flat panel lines (the eight-line vertical front porch for the current frame). The state machine then goes to the Sync Pulse state for four lines where it drives the flat panel vertical sync signal. It then goes back to the Nonsync state where it begins a new active line. This cycle repeats for seven flat panel display frames. On the eighth frame, the transition out of the Nonsync state goes to the Holdoff state. The state machine stays in Holdoff until VSYNC arrives from the input side. This is the signal to start the cycle again with another flat panel vertical sync. At this point, the input side and the flat panel display side of the frame buffer are synchronized.

Remember that one frame gets discarded. This frame is the first incoming frame after the synchronization event. Since the video input side has a 64-line vertical back porch, and the flat panel display side has none, the flat panel side will require a new frame of data immediately following the synchronization event, but the input side will not even start to write any valid data into the frame buffer until 64 horizontal lines later. In addition, since the input side is faster, it will catch up to the flat panel side sometime during this first frame causing a video tear. By not writing this first frame of data into the frame buffer, we can avoid the video tear. This is illustrated in Fig. 9. The numbers in the figure indicate

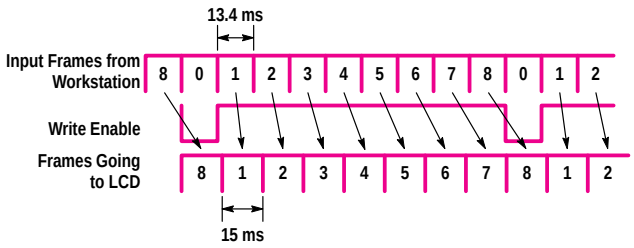


Fig. 9. Frame rate synchronization timing.

frames. Note that frame 0, which is the first frame in a sequence of nine frames, is not written into the frame buffer. The write enable signal controls writing a frame into the frame buffer. When low, the frame is skipped by not writing it into the frame buffer.

Acknowledgments

Many people made significant contributions to the design of this product. Bob Myers and Monish Shah gave valuable technical advice, John Metzner wrote the color centering software, Stuart Yoshida worked on the mechanical design, Howell Felsenthal managed the mechanical design team, and Paul Cacciola designed the power supply. Osamu Suzuki and Sunny Hattori were our communication channel to the flat panel vendors. Nancy Venturato provided marketing support, and Steve Grotheer and Tony Barton managed the unusual manufacturing requirements. The people who helped with field support, regulatory compliance, printed circuit board layout, and environmental test were critical, and unfortunately, too many to name. Last, but by no means least, we would like to thank Steve Becker, project manager for the electrical design team, and Mike Myshatyn, section manager for the electrical design team, for their support.

Applying an Improved Economic Model to Software Buy-versus-Build Decisions

The decision to buy or build software is a business decision that should be made using a sound economic model. A comprehensive economic model has been developed and applied to actual and estimated data to compare the costs of using a third-party software package to the costs of internal development.

by Wesley H. Higaki

Hewlett-Packard Laboratories has been recommending the use of third-party software packages to product divisions whenever it makes sense economically. We feel that those divisions whose primary products are not databases, networks, or operating systems should purchase these technologies. There are many readily available, shrink-wrapped, off-the-shelf products. Hewlett-Packard Company offers many application-specific software products such as electronic instruments and measurement systems. The product divisions that offer these applications can be more productive if they focus their efforts on the applications rather than devoting engineering resources toward developing technologies that are already available from third-party vendors.

There are, of course, valid reasons for not using third-party software in an application. Such software should not be used if the third-party product does not meet the application's specific functional requirements, if the third-party vendor is not reliable, financially stable, or responsive enough to customer needs, or if the third-party product costs too much.

While there are valid reasons not to use third-party software, there seems to be a bias in the engineering community towards building rather than buying software. The decision to buy or build software is a business decision that should be made using a sound economic model. Without this model and a complete analysis, inferior business decisions may be made.

This article applies a comprehensive economic model to actual and estimated data to compare the costs of using a third-party software package to the costs of internal development.

Past Economic Models

Today, when product teams do an economic evaluation to build or buy software, they sometimes make assumptions that can lead to less than optimal business decisions. Some of these assumptions are:

- *Initial software development costs are sunk.* The common view is that development engineers are already paid for, so it is always cheaper to write code internally than to buy it from the outside. Essentially, internal software development costs are considered zero. This is an indication that the return-on-investment model used today is flawed.
- *Ongoing maintenance costs are invisible.* Engineering effort will be expended over time to repair defects, add product enhancements and maintain compatibility with system upgrades. These costs, like initial development costs, are not visible and not accounted for in calculating returns.
- *Lost opportunities are not accounted for.* While development engineers are developing what could be bought, they are missing opportunities to add application value to the product. Development engineers may also help get their product to market earlier by not writing code from scratch.
- *Licensing costs are added to manufacturing costs and passed on directly to the customer.* This is a flaw in the software pricing model. If code is developed internally, the product is priced on market value. If code is bought and a license fee is attached, the licensing fee is generally considered a manufacturing cost.

An Improved Economic Model

To address the issues in the current economic models used, I propose the use of an improved model. There are four parts to this improved economic model for evaluating build-versus-buy decisions. These parts are:

- The Malan/Wentzel model
- An extensive list of costs and benefits of buying software
- Net present value calculations
- Estimation techniques for costs and benefits.

This improved economic model enables us to examine the hidden costs and benefits of buying software. We can then analyze the economic effects of buying more completely.

Malan/Wentzel Model

HP Laboratories has been researching models that evaluate the economics of software reuse. Using packaged software is a form of software reuse. The Malan/Wentzel model[1] produces a comprehensive picture of the true costs of software development and the savings that result from reuse. This

model is based on development phase costs, maintenance phase costs, and other reuse-related benefits and costs. Its developers point out that the savings in the development phase represent a conservative estimate of the benefits of reuse. The complete model takes into account the effects of increased profits resulting from early time to market and the exploitation of new opportunities.

List of Benefits and Costs

The following is a modified list from Poulin, Caruso, and Hancock[2] showing the benefits and costs of reusing packaged software. This list includes the benefits and costs of buying software through the development and maintenance phases. By using this list in conjunction with the Malan/Wentzel model, we can do a more complete analysis of the costs involved in making a buy-versus-build decision. In this discussion, a component is a piece of a software product whose functionality can potentially be filled by packaged software. This component may be purchased or it may be built.

Listed in Table I are the cost savings that may result during the initial development and maintenance phases if a component is bought rather than built.

Table I
Benefits of Buying

Initial Development Savings
Reduced cost to design the component
Reduced cost to implement the component
Reduced cost to test the component
Reduced cost to document the component
Ongoing Maintenance Savings
Reduced cost to fix defects in the component
Reduced cost to enhance the component

Buying software saves development time and thus the product can be delivered earlier than if all of the software is written from scratch. Earlier time to market can result in increased profits from two effects: added profit from delivering the product earlier to the marketplace, and added profit from increased market share over the life of the product.

It also costs something to buy software. The process involved in the build-versus-buy decision for software as described by Malan and Wentzel has four steps:

- Define the requirements of the component.
- Search for and acquire the component, determine what the component does, and verify that it meets the requirements.
- Integrate the component into the rest of the product.
- Customize the product to meet the specific application requirements.

There are costs associated with each of these steps relative to buying software. Table II shows a more detailed list of these costs.

Net Present Value Calculations

Since benefits and costs occur at different times in the product's life cycle, the time value of money must be taken into account. The net present value (NPV) equation[2] calculates the true net benefit of software reuse.

$$NPV = \sum_{i=0}^{n} (B_i - C_i)/(1 + k)^i,$$

where n is the number of years in the product's life cycle, k is the annual discount rate, B_i is the benefit realized in year i, and C_i is the cost incurred in year i.

Table II
Costs of Buying

Acquisition Costs
Licenses or royalties for the third-party package
Customization Costs
Cost of customizing the third-party package
Cost of maintaining the customized component
Assessment and Integration Costs
Cost of performing the cost-benefit analysis
Cost of locating and assessing the third-party package
Cost of integrating the third-party package
Cost of training on the third-party package

Benefits of Buying

Initial Development Savings. The most immediate benefit of buying software over building it is that the initial engineering effort and costs are saved. This saving can be measured directly, based on the estimated engineering cost to build the component. This cost can be estimated using schedule estimates to design, develop, and debug the component multiplied by the run rate for the engineers involved.

Ongoing Maintenance Savings. Harris[3] estimates that software maintenance costs for internally developed software are about 55% of the total cost of the product life cycle. This estimate includes costs associated with fixing defects as well as implementing enhancements. We can estimate the reduced maintenance costs using the actual initial development costs of the internally developed software by the following formula:

$$C_M = (0.55/0.45)C_D,$$

where C_M is the cost of maintenance over the life of the product and C_D is the initial development cost of the product. Thus, using the estimate from the initial development, we can calculate the expected ongoing maintenance costs.

Increased Profits. Patterson[4] states that for every month a product release is delayed, one month of sales is lost. Conversely, delivering a product early results in increased sales. Patterson asserts that once a product with a given set of capabilities is released, the date the product will become obsolete is set. Thus, if two products are released at different times with the same capabilities, they will both be obsolete on the same date because of marketplace and competitive pressures. If these two products are released on different dates, then the product released earlier will have the greater sales. Fig. 1 illustrates the effect of an early

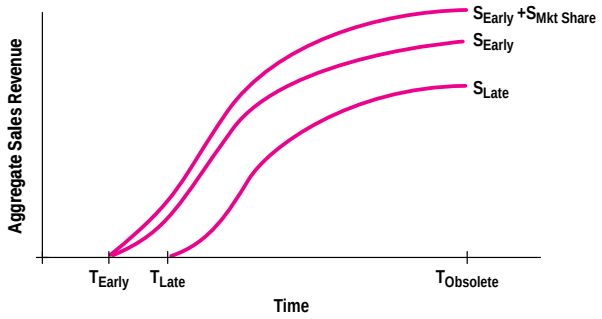


Fig. 1. Effect on sales of earlier time to market. Curve S_{Early} represents the sales of the product introduced at time T_{Early}. The curve S_{Late} is the sales curve for the product introduced at time T_{Late}. The curve $S_{Early} + S_{Mkt\ Share}$ illustrates the total effect on sales of more time in the marketplace and increased market share.

product introduction on sales revenues. Curve S_{Early} represents the sales of the product introduced at time T_{Early}. The curve S_{Late} is the sales curve for the product introduced at time T_{Late}.

Software reuse can improve a product's time to market. If the product reaches the market two months early with the same functionality, then there will be an additional two months' revenue. To estimate this effect, we will use the projected annual sales and estimate the improvement in time to market.

$$S_I = S_V T_M,$$

where S_I is the sales increase resulting from early time to market, S_V is the original projected volume per month, and T_M is the improvement in time to market in months.

Moreover, Smith and Reinertsen[5] suggest that market share increases over the life of the product as a result of earlier time to market. This is expressed as an increase in market share percentage throughout the life of the product. This estimate is based on competition, market demand, and time to market. This effect can be estimated by using market research data to determine how sales volumes would be affected by providing features earlier than the competition. This can be expressed as more units sold per year or per month. The curve $S_{Early} + S_{Mkt\ Share}$ in Fig. 1 illustrates the total effect on sales of more time in the marketplace and increased market share.

Costs of Buying

Acquisition Costs. Sometimes there are up-front acquisition costs for third-party software. More typically, there are licensing costs or royalties associated with using a software package. These costs can be easily estimated by using sales volume estimates for the end product and determining how many copies of the third-party package will be required.

Customization Costs. This model assumes that no modification of the third-party software is required to meet the product requirements. However, customization may be required. An example of such customization is using Microsoft® Excel to build a spreadsheet that calculates car payments. The customization is the act of developing the spreadsheet to do the appropriate calculations. The Excel product itself is not modified. The customization costs are derived by estimating the engineering effort for customizing the third-party package.

Assessment and Integration Costs. To determine if in fact a third-party package will meet the product requirements, some engineering evaluations must take place. Estimates of how much effort and cost are involved in the evaluation, assessment, and integration of the third-party package into the application can be made.

Case Study

In a project that HP Laboratories recently completed with an HP product division, the product team made extensive use of third-party tools and components, but chose to build a report writer subsystem rather than buy a report writer package. The product required that reports be generated from data stored in a dBASE IV® database on a PC running Microsoft Windows 3.1. Several off-the-shelf packages were available ranging in price from $200 to $2,500 per copy (all amounts in this paper are in U.S. dollars). One package seemed to meet the requirements and cost $400 per copy, but the product team decided to develop the report writer internally. This decision was made based on the assumption that $400 per copy was more than what it would cost to build the report writer internally.

The report writer is only one component in the product. If the recommended third-party package had been used in place of the internally developed report writer, the project would have been completed about two months earlier. This third-party package is a tool that enables the developer to create custom reports from data stored in dBASE IV. It would take an engineer about one week to learn this package, another week to develop the report formats, and one more week to integrate it into the rest of the product. An ongoing enhancement effort of about one week per year to add new report formats is expected to continue throughout the product's life.

Table III summarizes the costs and benefits determined using the economic model to compare the use of this third-party package with an internally developed report writer. According to the model, the net savings of buying the software in this case would have been $213,754 over the life of the product.

Initial Development Savings. The actual time it took to design, implement, integrate and test the report writer for this project was one engineer-year. A fully loaded engineer costs the company about $100,000 per year. So, the savings in initial development would be $100,000. These savings would have been realized in the year before the product release.

Ongoing Maintenance Savings. Using the maintenance cost estimate formula, the estimated maintenance cost for the lifetime of the product is:

$$C_M = (0.55/0.45)(\$100,000) = \$122,222.$$

Assuming this cost is evenly distributed over the four-year life of the product, the cost is about $30,500 per year.

Introducing the time value of money at a 6% discount rate results in a $105,686 benefit.

Table III
Cost-Benefit Analysis

Benefits	Savings or (Costs)
Initial Development Savings	$100,000
Reduced cost to design	
Reduced cost to implement	
Reduced cost to test	
Reduced cost to document	
Ongoing Maintenance Savings	$105,878
Reduced defect fixing costs	
Reduced enhancement costs	
Increased Profits	$72,387
Added profit from delivering product sooner to the market	
Costs	
Acquisition Costs	($41,581)
Licenses or royalties for reusing parts	
Customization Costs	
Cost of customizing the third-party package	($2,000)
Cost of maintaining the customized component	($6,930)
Assessment and Integration Costs	
Cost of performing the cost-benefit analysis	($5,000)
Cost of locating and assessing the third-party package	($5,000)
Cost of integrating the third-party package	($2,000)
Cost of training on the third-party package	($2,000)
Net Savings	$213,754

The maintenance effort includes not only defect fixes,[6] but also updates for new fonts, printers, and report formats. The internally built report writer was only a set of report templates and not a template generation tool like the recommended third-party package. This means that whenever there is a request for a new report template, more software needs to be written. With the third-party package, a new template can be created quickly. Since much of the division's sales are for custom systems, the Harris estimate[3] seems to be consistent with this example.

There are also quality and competitive issues involved with developing an internal report writer. Since the third-party company is in the business of developing and selling report generators, they (and their competitors) will invariably do a more complete job of developing report generators than this HP division will. The HP division will be forced to compete with them as customers demand greater functionality to match that offered in packaged report writers and this will increase the enhancement effort.

Increased Profits. Based on projected sales for this product and the anticipated earlier product introduction, the division would realize $500,000 additional revenue. It is also estimated that there would be more units sold per year through out the entire product life because of the earlier initial product introduction and the availability of engineering resources to develop value-added features. Using an appropriate profit margin on the increased sales volume and applying the NPV calculation yields the $72,387 benefit shown in Table III.

Acquisition Costs. There are no fees associated with the initial acquisition of the third-party product, only licensing fees of $400 per copy. Applying the NPV formula to the estimated unit sales for the life of the product results in a $41,581 cost to pay for the licenses.

Customization Costs. While no changes need be made to the third-party package, report templates would have to be generated for the HP product. The effort required to generate the initial report templates and to integrate the report generator with the rest of the application is estimated to cost about $2,000. The engineer would have to develop the report formats using the third-party package. It is also estimated to cost about $2,000 per year to add any new report formats requested by customers. The NPV formula is applied to these costs for the life of the product to yield $6,930.

Assessment and Integration Costs. It is estimated that an engineer would cost $10,000 to do the domain analysis, locate and assess the packaged software, and do a cost-benefit analysis. It is also estimated to take another $2,000 to integrate the package into the rest of the HP product.

The cost of learning the third-party product is estimated to be about $2,000.

Cost-Benefit Analysis. First, we compare just the costs of development with the licensing costs. The total cost of developing and supporting the report writer is about $206,000 ($100,000 initial development plus $106,000 for ongoing maintenance). Divide this by the number of units expected to be sold during the product's life, and the cost per copy of this report writer is estimated to be over $1,000. These costs far exceed the $400 per copy licensing fee for the third-party package. This alone indicates a decision to buy rather than build.

To do the complete analysis, increased profit factors and package assessment costs are introduced. The net benefit of using the third-party package rather than developing a report writer is $213,754.

The greatest benefit comes from delivering the product early. Arguments from Stalk and Hout,[7] Patterson,[4] and Smith and Reinertsen[5] list the advantages of delivering products early, and this model demonstrates the impact time to market has on a product's revenues. The time-to-market benefits dwarf the $400 per copy licensing costs of the packaged software.

Conclusions

The product team thought they would have to add the $400 cost of the third-party product as a separate line item on the customers' purchase orders. They were concerned that this would confuse the customer, causing discussions that could hinder the sale and reduce overall sales. If the third-party package were bundled with the product so that the customer could not see what was included in the product, there could have been no objections. For accounting purposes, this $400 cost could have been recorded as an R&D expense and not a manufacturing expense since manufacturing expenses automatically have multipliers attached. The total

price would remain the same, but profit would be greater because costs would be lower.

Risks

While the focus of this article has been on the cost-benefit analysis of buy-versus-build decisions, this analysis is useful only after the preliminary evaluations have taken place and the package has been proven to meet the requirements of the product. These requirements must be carefully evaluated first.

Product quality and reliability are especially important to HP, a company for which quality is a major differentiator in the marketplace. The packaged software must meet the functional and quality requirements before any further consideration. Using a packaged product that comes from a proven vendor and has a history in the marketplace of providing complete functionality and solid support mitigates the risk of choosing an inadequate package.

Another major risk is the longevity of the vendor. No one wants to rely on a package that is no longer supported by the vendor. Again, using packages from proven vendors will help reduce this risk. Also, choosing components that are supported by a number of qualified packages from competing vendors allows the replacement of one package with another in case the initial vendor cannot maintain the package any longer.

There are several risks involved in developing software internally. The more obvious risk is the effort required for defect fixing. There is also the risk that internally developed software will have to compete with other packaged software. This is a battle that cannot be won, since report writer vendors, for example, expend much more effort developing report writers than does a product team trying to deliver a measurement system.

Acknowledgments

Many thanks to Ruth Malan of HP Laboratories, who has encouraged me, helped direct me to the proper sources, and provided inputs on the approach to take in presenting my ideas.

References

1. R. Malan, K. Wentzel, *Economics of Software Reuse Revisited*, HP Laboratories Technical Report HPL-93-31, April 1993.
2. J.S. Poulin, J.H. Caruso, and D.R. Hancock, "The Business Case for Software Reuse," *IBM Systems Journal*, Vol. 32, no. 4, 1993.
3. K. Harris, "Using an Economic Model to Tune Reuse Strategies," *Proceedings of the Fifth Annual Workshop on Software Reuse*, 1992.
4. M. Patterson, *Accelerating Innovation*, Van Nostrand Reinhold, 1993.
5. P.G. Smith, and D.G. Reinertsen, *Developing Products in Half the Time*, Van Nostrand Reinhold, 1991.
6. W.T. Ward, "Calculating the Real Cost of Software Defects," *Hewlett-Packard Journal*, Vol. 42, no. 4, October 1991, pp. 55-58.
7. G. Stalk, Jr. and T.M. Hout, *Competing Against Time*, The Free Press, 1990.

Benchmark Standards for ASIC Technology Evaluation

Two benchmark circuits are used for objectively evaluating ASIC supplier performance claims. The method applies first-order equations relating capacitive discharge currents and transistor saturation current to arrive at a technology constant. The method has been used to survey 14 ASIC suppliers with over 76 different technologies. Results are shown for 48 CMOS technologies.

by Antonio A. Martinez, Aloke S. Bhandia, and Henry H.W. Lie

The issue of determining valid performance for ASIC suppliers is always paramount in the minds of designers. Choosing a supplier based on aggressive performance claims could lead to disastrous results. On the other hand, choosing a supplier with conservative performance claims results in higher costs than necessary because of suboptimal performance and area utilization.

One way to compare ASIC supplier technology performance is through benchmark circuits. The traditional simplistic 2-input NAND gate intrinsic delay or even fanout delay has given way to more complete benchmark circuits. But which benchmark circuit covers the range of design possibilities? Should it scale with technology parameters such as transistor drive, metal wire length, or capacitance? What about interconnect metal resistance? What are appropriate interfaces to the real world? TTL or CMOS? Beyond design considerations, what conditions were used to generate performance and delay numbers? What voltage, temperature, input slew time, and process conditions were used by the ASIC supplier? Our investigation addresses these issues.

After contacting several HP divisions, we ended up using two benchmark circuits that have been around in various forms since 1987 and were most recently published in 1993.[1] We surveyed ASIC technologies from various suppliers to compare benchmark circuit performance claims. Since the benchmark simulations give too much latitude to ASIC suppliers, we used specific device technology details to compare and evaluate the accuracy of supplier claims. We observed both aggressive and conservative performance claims given the underlying technology. The process we describe also allows HP designers to focus on areas where a supplier has suboptimal designs.

Description of the Benchmarks

The benchmark circuits are shown in Fig. 1. The first circuit, Benchmark 1 (Fig. 1a), shows a complete path from input pad to output pad. There is a substantial output pad load of 50 pF which more closely represents real-world applications. An internal path contains various typical logic gates with fanout and wire length specifications. The second circuit, Benchmark 2 (Fig. 1b), includes two D flip-flops and a 2-input NOR gate. Benchmark 2 does not include interfaces to the real world and is relatively simple in comparison to Benchmark 1. To a larger extent, Benchmark 2 gives a closer feeling for the intrinsic performance of a given technology. For both benchmarks, wire length and gate fanout capacitance scale with technology. For instance, if metal pitches become smaller when moving to the leading-edge technology, wire lengths should be shorter. Likewise, gate fanout capacitance should track when moving to smaller, stronger transistors in a leading-edge technology.

We obtained complete path delays for 48 CMOS technologies for 14 different ASIC suppliers, including internal and I/O path rising and falling delays for Benchmark 1 and maximum operating frequency for Benchmark 2. Generally, suppliers used models of their technology to estimate path delays rather than measurements of actual circuits. The data collected is shown in Table I.

Table I
ASIC Supplier Sample Data

Drawn transistor gate length (μm)
Effective transistor gate length (μm)
Transistor gate oxide thickness (Å)
Pitch for each metal layer (μm)
Power supply voltage used for benchmarks (V)
Junction temperature used for benchmarks (°C)
Process condition used for benchmarks (SLOW/NOM/FAST)
Input edge rate used for benchmarks (ns)
Benchmark 1 complete path T_{pLH} delay (ns)
Benchmark 1 complete path T_{pHL} delay (ns)
Benchmark 1 internal path T_{pLH} delay (ns)
Benchmark 1 internal path T_{pHL} delay (ns)
Benchmark 2 maximum operating frequency (MHz)

Suppliers were required to provide enough process details to evaluate the accuracy of performance claims. We wanted to verify performance using first-order figures of merit and comparisons between suppliers. To that end, we considered process and device parameters that strongly affect circuit performance, namely transistor effective gate length (L_{eff}),

transistor oxide thickness (T_{ox}), junction temperature, power supply voltage, and pitch for all metal layers. These parameters are shown in Table I.

Evaluating Performance Claims

We evaluate supplier performance claims by relating delay or frequency numbers to manufacturing process specifics. Roughly speaking, propagation delay t_d is proportional to L_{eff}, T_{ox}, and temperature, and inversely proportional to V_{DD}, as shown in equation 3 below. We arrived at equation 3, a first-order constant relationship, by equating the rate of capacitive discharge current (equation 1) to transistor saturation current I_{DSAT} (equation 2). Equation 1 models the capacitive discharge of a node in terms of the power supply voltage, current, and rate of discharge (roughly proportional to delay). Equation 2 relates transistor saturation current to transistor effective gate length L_{eff}, oxide thickness T_{ox} (inversely proportional to oxide capacitance $C_{ox)}$, mobility u_o (related to temperature in Kelvin), threshold voltage V_{to}, transistor width W_{eff}, and power supply voltage V_{DD}. For simplicity, V_{to} is assumed to be proportional to $V_{DD,}$ while temperature is inversely proportional to u_o.

$$I = C\frac{dV}{dt} \tag{1}$$

$$I_{DSAT} = \frac{u_o C_{ox} W_{eff}}{2\ L_{eff}}(V_{DD} - V_{to})^2 \tag{2}$$

$$\text{Technology Constant} \approx \frac{t_d V_{DD}}{L_{eff} T_{ox} \text{Temp}}. \tag{3}$$

Equation 3 is not very precise, and is valid only over narrow regions of operation. However, detailed SPICE simulations showed that there is a surprising amount of linearity, as shown in Fig. 2. The largest error was 6.4% for L_{eff} variations from 0.4 µm to 0.9 µm. The largest percent errors are shown in Table II. Simulations were done with HP SPICE for a circuit deck equivalent to Benchmark 2 using 0.5-µm CMOS SPICE models.

Table II
Maximum Percent Error for Linear Fit of Delay as a Function of Various Variables

Variable	Range	Maximum Error
L_{eff}	0.4 µm to 0.9 µm	–6.4%
T_{ox}	57 nm to 228 nm	–4.0%
Temperature	0°C to 135°C	–1.2%
V_{DD}	2.7V to 5.7V	5.8%

Supplier Performance Comparisons

We first charted all performance numbers for various suppliers as shown in Fig. 3. For each supplier, L_{eff} decreases as we move from left to right. Suppliers are generally quite competitive with their leading-edge technology, but some suppliers' leading-edge technologies appear slower than another supplier's mature technology. Some suppliers appear to do quite well in the total path delay for Benchmark 1 but do not perform as well in the maximum operating frequency for Benchmark 2.

We applied equation 3 to all delay and frequency numbers, computing a *technology constant* value for each supplier's technology performance claim. However, since the technology constant is only a constant under an idealized first-order approximation, we looked for patterns and trends. We computed the mean and standard deviation for all suppliers across all technologies for each Benchmark 1 and 2 constant. After plotting technology constants for each benchmark for all suppliers and their technologies, we placed guardbands one standard deviation from the mean as shown in Fig. 4. Technology constants falling above the upper guardband were labeled conservative and technology constants falling below the lower guardband were labeled

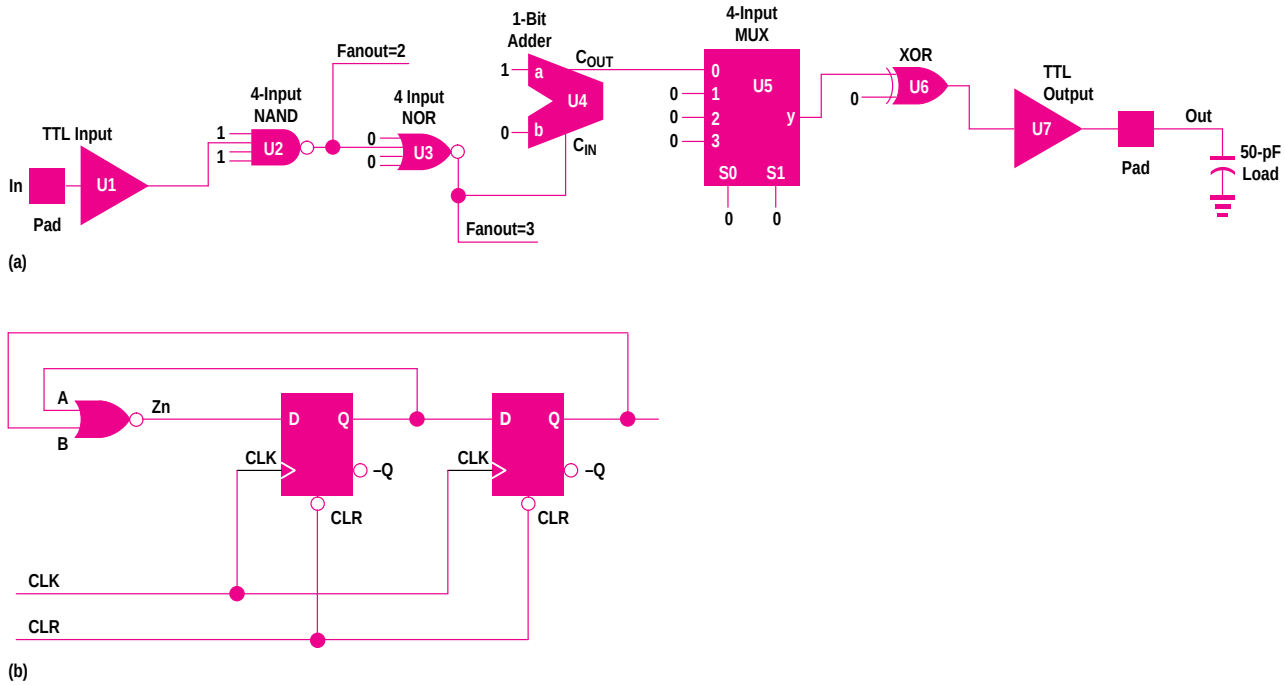


Fig. 1. (a) Benchmark 1 includes a complete path of internal and I/O delays. (b) Benchmark 2 includes D flip-flops with reset.

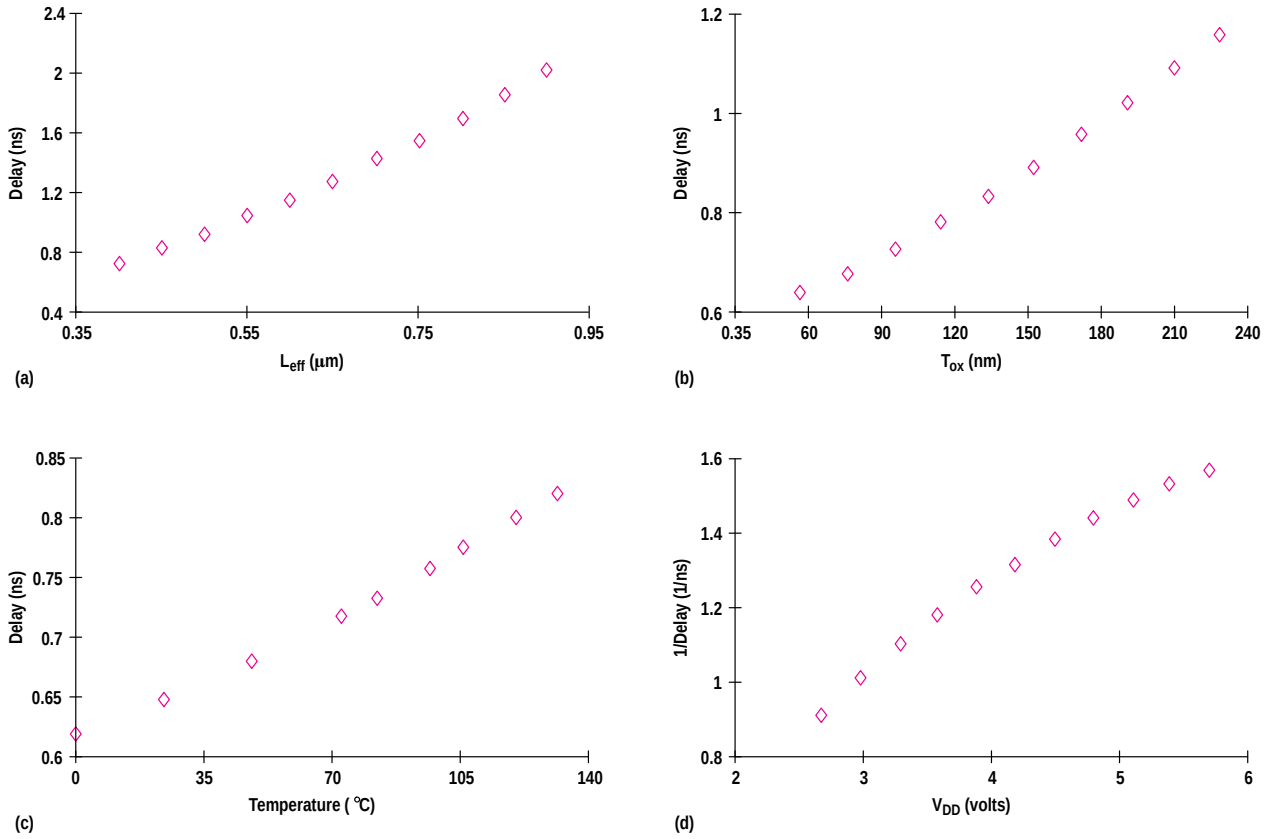


Fig. 2. SPICE plot of delay as a function of (a) L_{eff}, (b) T_{ox}, (c) temperature, and (d) V_{DD}.

optimistic. It should be noted that technology constant values are different for different benchmarks or portions of a benchmark.

From an examination of Fig. 4, some trends and patterns emerge. For instance, many suppliers tend to be more conservative for their leading-edge technology. This might result from inherent tendencies to be more conservative. It could also be that transistor performance, as measured through electron velocity saturation, will tend to level off for smaller L_{eff}.

Some suppliers clearly claimed performance that is simply unattainable given their technology description as seen in Fig. 4. As we investigated further, we found out why they appeared conservative or optimistic. For instance, supplier G used a 2-input NAND gate instead of a 4-input NAND gate, claiming that they were optimizing the critical path using a 2-input NAND gate for the critical path and a 3-input AND gate for the rest of the noncritical path. Supplier K used half the typical unit wire capacitance constant for computing internal net capacitances: 0.1 fF/µm instead of 0.2 fF/µm. On the other extreme, supplier F, while having competitive performance, clearly was capable of doing substantially better. After detailed conversations, it became apparent that their technology is immature and poorly defined, with high net

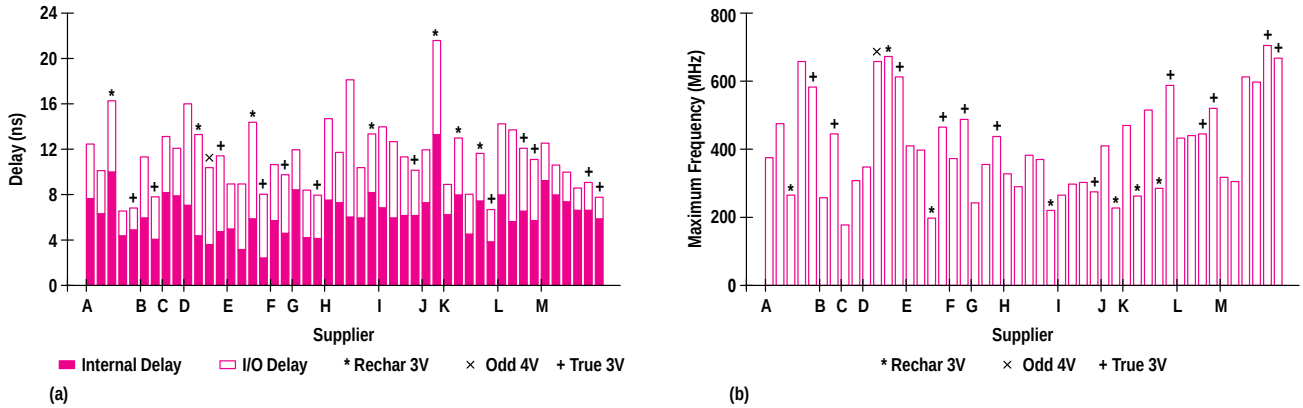


Fig. 3. Total path delay including (a) Benchmark 1 internal plus I/O delay and (b) Benchmark 2 maximum frequency for various technologies and suppliers. For each supplier, L_{eff} decreases from left to right for their respective technology offerings. Some technologies are simply recharacterizations from 5V to 3V (*) while others are true 3V (+).

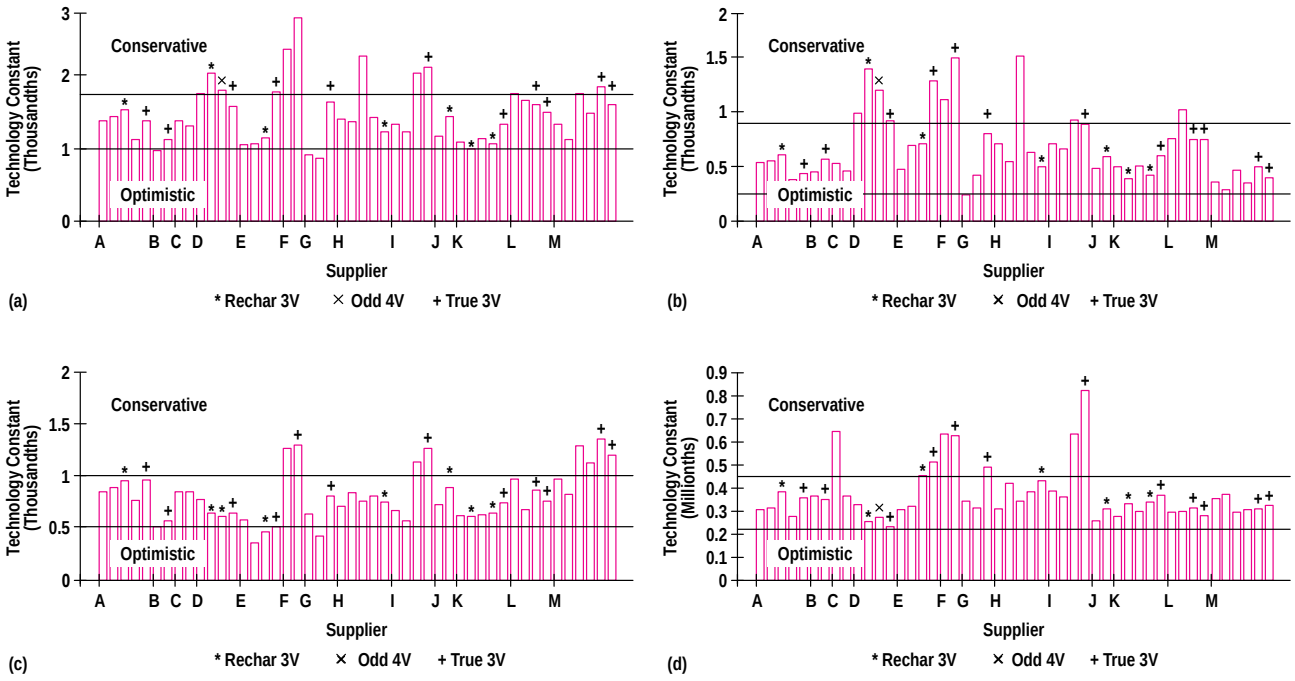


Fig. 4. Optimistic versus conservative technology constants for (a) Benchmark 1 total path delay, (b) Benchmark 1 I/O path delay, (c) Benchmark 1 internal path delay, and (d) Benchmark 2 maximum frequency for various technologies and suppliers. For each supplier, L_{eff} decreases from left to right for their respective technology offerings. Some technologies are simply recharacterizations from 5V to 3V (*) while others are true 3V (+).

capacitances computed based on previous technology offerings and less than compact layout. Comparisons through the technology constant method also allow us to determine if suppliers have potentially optimized I/O or internal cells as seen in Figs. 4b, 4c, and 4d, respectively.

Table III compares supplier K to another with very similar gate array technology, supplier H. Both suppliers offer gate array products. Supplier K used a wire capacitance constant of 0.1 fF/μm, which explains partly why they claimed higher performance. In fact, supplier K has larger metal pitches than supplier H, which should result in substantially longer net wire lengths after place and route, further increasing net wire capacitances and decreasing overall performance. Both suppliers use Cadence's Gate Ensemble for layout, so it is unlikely there would be substantial wire length differences even if they had the same metal pitches. As stated previously, supplier K was found to have overly optimistic performance claims. Comparisons through the technology constant method allowed us to home in quickly on the reasons why, chief among them being the unrealistically low wire capacitance constant of 0.1 fF/μm.

Table III
Comparison of Supplier Technologies

	Supplier	
	H	K
Effective transistor gate length(μm)	0.7	0.7
Transistor gate oxide thickness (Å)	150	150
Pitch for each metal layer (μm)	2.0,2.8,2.8	3.2,3.2,3.2
Power supply used for benchmarks (V)	4.5	4.75
Junction temperature used for benchmarks (°C)	85	85
Process condition used for benchmarks (SLOW/NOM/FAST)	SLOW	SLOW
Input edge rate used for benchmarks (ns)	1	1
Benchmark 1 complete path T_{pLH} delay (ns)	14.4	8.04
Benchmark 1 complete path T_{pHL} delay (ns)	12.2	8.72
Benchmark 1 internal path T_{pLH} delay (ns)	7.3	4.8
Benchmark 1 internal path T_{pHL} delay (ns)	7.1	4.4
Benchmark 2 maximum operating frequency (MHz)	317	460

Assumptions and Future Extensions

For the sake of expediting the process of ASIC technology benchmarking, we used a simplified approach that we can improve upon for future supplier analysis. Our detailed questions did not ask for wire unit capacitance; one supplier used wire capacitance values half those given by conventional wire capacitance statistical modeling. Neither did we include wire unit resistance effects. However, the benchmarks have relatively short wire lengths so it is unlikely that RC delays contribute significantly to total delay. In the future, we can specify gate array and standard cell height as

the unit of length. We can also include the effects of transistor mobility, in particular threshold drops (V_{to}), which become important as we scale down to lower power supply voltage levels.

We also should specify more precise input signal edge rates, power supply voltage, and junction temperature, such as 1-ns 10-to-90% rise and fall times, 10% off nominal power supply (4.5V for 5V or 3.0V for 3.3V), and 85°C junction temperature. The benchmarks left these numbers to the discretion of each supplier. Sometimes junction temperatures varied from 70°C to 125°C while power supply voltage varied from 5% to 20% off nominal. In the near future, we want to get supplier electrical and SPICE models, and ultimately verify performance through actual silicon.

Equation 3 is only a first-order approximation. Curve fit analysis shows the best fit for the SPICE simulation data is not necessarily always linear. Equation 3 might be better evaluated as equation 4 below for some situations:

$$\text{Technology Constant} \approx \frac{t_d \ln(V_{DD})}{(L_{eff})^{1.25} e^{T_{ox} \times \text{Temp}}} \quad (4)$$

We are also currently evaluating other issues such as optimal metal pitch as a function of transistor L_{eff}, and library richness. Although having smaller metal pitches is advantageous in terms of routability and increased interconnection, there is a balance between transistor on-resistance and metal wire resistance, and between routing density and process cost and manufacturability. If wire pitch is too fine, wire resistance will dominate over transistor output drive. Furthermore, finer metal pitches increase process cost and reduce yield as well as long-term reliability.

Library richness is another area considered critical by many designers. In particular, designers want a rich and complete set of library functions to allow maximum flexibility in implementing chip designs. The library must be well-modeled and compatible with various CAD tools, especially mainstream synthesis and simulation tools. This includes optimizing cell drives and functionality for synthesis.

In the future, we need to review synthesis and simulation libraries for a number of ASIC technologies using small to medium-size benchmarks. Critical path timing and gate count should be evaluated after synthesis. We need to have commonly agreed-to benchmarks to evaluate and compare all major ASIC supplier libraries. These benchmarks should cover areas such as scan insertion, error correction and detection, RAM models, and others in addition to critical path timing and area optimization. The benchmarks should not include HP proprietary information so they can be freely used with external suppliers. This allows suppliers to run evaluations using their resources. HP divisions would simply have to corroborate ASIC supplier results. The benchmarks should be available in both VHDL and Verilog hardware description languages since both are being used within the HP community.

Part of the survey asked for detailed power dissipation for various portions of Benchmark 1 and Benchmark 2. Often, simulated power dissipation numbers were significantly out of line with common sense analysis, giving us an indication of the limitations of power estimation CAD tools for some suppliers. We are investigating the area of power estimation as it relates to mainstream CAD tools and supplier methodologies.

Conclusions

We have developed a simple, first-order method for quickly determining the degree of optimism or conservatism of ASIC technology performance claims from various suppliers. We found suppliers that are not capable of delivering performance as promised because of circuit tricks played with the benchmark or too-aggressive wire capacitances (0.1 fF/μm instead of 0.2 fF/μm). We also found suppliers that may have immature, poorly defined technology and design libraries.

Our method helps designers choose appropriately characterized ASIC technology while avoiding the disastrous consequences of choosing an ASIC supplier not capable of delivering the promised performance. On the other hand, it points out inefficient or immature suppliers that may be incurring extra costs because of suboptimal utilization of circuit performance and area. The technology constant method also allows us to identify suppliers with potentially superior or optimized I/O or internal design libraries.

The method is adaptable to any number of circuit benchmarks and ASIC suppliers. However, it is only a first step in the process of evaluating ASIC technologies. There are many other factors that should be considered when selecting ASIC technologies.

Acknowledgments

The authors would like to thank Hyoung Jo Youn for contributing his CMOS26 and CMOS14 SPICE decks. Also, thanks to Robin Purohit and Noori Din for discussions and enlightenment on the issues of library evaluation through synthesis. Our heartfelt gratitude goes to Sharon Fingold, wife of one of the team members, whose keen eye and clear mind kept our ideas and writing focused. Finally, we would like to mention the names of many other dedicated and creative people who contributed directly or indirectly through their discussions and otherwise, including but not limited to: Fritz Stafford, Harold Noyes, Theresa Prenn (Tideswell), Greg Marten, Bruce Schober, Rob Mankin, Scott Trosper, Gulu Advani, Atul Goel, John Jacobi, Donovan Nickel, Steve Chalmers, Margie Evashenk, Mike Stewart, Dave Fotland, Dave Goldberg, Bob Schuchard, Todd Spencer, Jon Riesberg, David Stewart, Paul Dembiczak, David Bond, Jeff Dawkins, Oren Rubinstein, and Alex Warshawsky.

Reference

1. L. Vereen and M. Jain, "Benchmarks Define ASIC Performance," *ASIC&EDA*, September 1993.