Datapoint Announces New Processor

In October, Datapoint® announced the 1560 Processor, the newest member of the 1500 series, designed for use in data entry, data processing, word processing, electronic message service, and data communications applications in the small business or professional workstation environment.

The 1560 is an enhancement to Datapoint's initial 1550 and is compatible with the 1500 processor[1]. The 1560 supports Datapoint's DATABUS® business programming language, FORTRAN, and BASICPLS. The processor can function either as a standalone system or as the host processor in a four-workstation DATASHARE® configuration and (optionally) even as part of an ARCNET™ local area network, allowing up to 255 DOS.H processors to be linked together to form a local network.

Datapoint 1560

System Capablities

A 1560-based Attached Resource Computer® (ARC™) system can coexist as part of a conjoined network that includes a complete Datapoint Resource Management System™ (RMS™) ARC network. The

Datapoint's ARC Keeps Customers Happy

When Datapoint advertisements say "We make systems that work together now," it's not just a slogan. Here at Datapoint, one network works with another to keep the company up and running. For example, do you ever wonder just how you get consumables and supplies when you call Customer Service with an order? There's no magic involved; rather there is a network that fills not only your orders, but orders for all Datapoint needs as well. The Logistics Department at Datapoint uses part of a large Attached Resource Computer (ARC) network that communicates with local buildings, field offices, and remote vendors.

The Logistics Department and Its Growth

The Logistics Deparment is committed to customer satisfaction. One of the deparment's objective is to take customer orders for consumables (like ribbons) and supplies (like paper) and ship those orders in a timely, accurate manner. The department also wants to be able

1560 can also function under Datapoint's Disk Operating System (DOS.H) or CP/M* software. Word processing is available by using Datapoint's Integrated Electronic Office™ software. Messages can be sent and received electronically with other Datapoint equipment using the Electronic Message System.

CP/M and the 1560

One of the key features of the new 1560 is its CP/M capabilities. Control Program/Microprocessor (CP/M) is an operating system owned, copyrighted, and trademarked by Digital Research, Inc. Over the past ten years, CP/M has become the standard operating system for 8080- and Z80-based microcomputers. Manufacturers of this class of equipment have almost universally offered CP/M as the primary operating system. Although CP/M must be modified for each new computer that uses it, the modification is so straightforward that each manufacturer typically performs the modifications for his own equipment.

The 1560 will eventually run with standalone CP/M applications and will later tie into the ARC (using the 1560 file processor), providing integrated and controllable growth paths for systems and data bases.

Other Features

Package programs are available from Datapoint for use on the 1560, and Multiplan* is just one example of many. IBM mainframe communications and terminal emulation can be performed with Datapoint's EM3270 software package, not to mention other batch and interactive software emulator packages. SDLC communications discipline support is available with the Multifunction Communications option on the 1560.

A variety of Datapoint disk and diskette drives provides storage for the 1560 processor. Double-density diskette storage is available in 1MB and 2MB configurations with the 1403 and 1404 Diskette Drives, respectively. The 9313/9315 Disk Drives provide 5 or 10MB of on-line storage with a 1MB diskette for backup; the 9310 Cartridge Disk Drive provides 10MB of online storage or

backup. Four disk or diskette drives can be supported on the microbus for a maximum total of 40MB of on-line, fast-access data storage.

For further information concerning the new 1560 processor and its capabilities, please contact your local Datapoint representative.

[1] Existing 1500 and 1550 Processors can be field-upgraded to support the features of the enhanced 1560.

*CP/M is a registered trademark of Digital Research, Inc. Multiplan is a trademark of Microsoft, Inc.

The ARCNET capability of the 1560 is currently under development and is planned for availability in the second calendar quarter of 1983.

1560 System Architecture
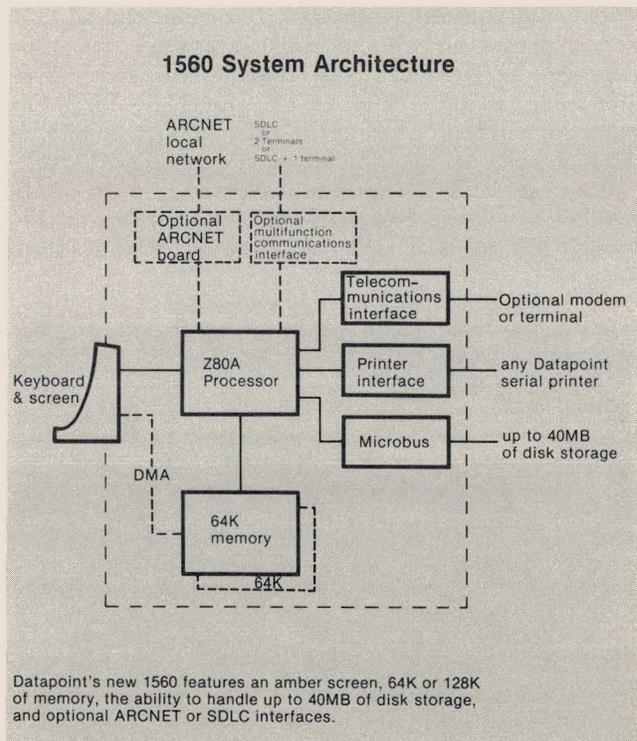
Datapoint's new 1560 features an amber screen, 64K or 128K of memory, the ability to handle up to 40MB of disk storage, and optional ARCNET or SDLC interfaces.

---

to answer any questions the customer may have, such as "When was my order shipped?" and "Exactly what was shipped in that order?"

The present network has grown out of the need to fulfill these requirements and to meet the increased demand for supplies that has evolved over the past few years. Before it was in operation, another, smaller ARC required that all orders for consumables and supplies be taken by Customer Service, put on a cassette tape, delivered to Software Services by courier, and finally read and filled by a clerk: a time-consuming and inefficient way of meeting the customer's daily needs. Eventually, Customer Service began to split orders into those for software and documentation and others for consumables such as paper. The orders for consumables and supplies were sent via DATAPOLL® from the Customer Support Center to Logistics, though orders for software and documentation were still being sent to Software Services on a cassette. When the number of customer orders slowed the efficiency of this network, the need for a more streamlined network became apparent.

Networking the Networks

The network was changed about a year ago, when the ARC in Customer Service began communicating with the ARC in Logistics. The two ARC networks handle the order input and inquiry process in tandem: at the Customer Support Center, the coordinator takes the call; in the Logistics Department, the order is printed, material is picked, packed, shipped, and waybill information is recorded to assist in customer inquiries and shipment tracking. The networks communicate with each other through multiplexed lines, DATASHARE links, ARCLINK™, and DATAPOLL. Even when this larger network became functional, the way of doing things still needed improvement. In the network's early days, clerks were spending two to three hours per day or more on the phones getting orders to vendors. Besides having a huge phone bill, the number of errors was unacceptable. With a customer base that needed orders filled daily, major improvements were mandatory to streamline the order handling procedures. One glaring fact stood out: the present network just wasn't doing the job.

Eventually, by finding ways to use the network to its fullest potential, orders were sent in the correct format to the Logistics Department. Those orders could then be sent directly to remote vendors who supply products for Datapoint. The orders are sent over communications links or by mail. So, within a year of its inception, the entire system had clear, direct impact on the departments involved: clerical time was cut by at least 25%, orders are initiated more quickly, and errors are almost eliminated. Most importantly, all of these improvements have increased customer satisfaction.

The ARC In Logistics

In the Logistics Department, the ARC is composed of several "subnetworks" such as the Emergency Order Center and the Warehouse. These departments run several different types of programs, using the full range of Datapoint's products. Multiplexed lines connect the departments to the ARC in Customer Service. DATABUS programs run the networkmanagement information in the network can be transferred to other areas using DATASHARE programs, or sent to remote vendors via the Electronic Message Service, DATAPOLL, and other communications disciplines.

General Flow of Supplies Processing

Logistics Control System (LCS)

Field Inventory Control System (FICS)
Inventory Control System (ICS) ARC

20 major parts depots in the U.S.,
each with an 8200 terminal,
212A Bell modem, and printer

TYMNET
Timesharing network
on telephone

Direct Line

ARC continued from page 3

The ARC in the Logistics Department not only provides links between Customer Service and remote vendors, it also allows Datapoint parts depots across the United States to communicate with the Logistics Department to order parts, update inventory records, and locate needed parts. TYMNET, a timesharing network Datapoint uses, provides this data communications link between the parts depots and the Logistics Department.

An Emergency Network

Twenty parts depots are located throughout the United States to service customers in every geographic area. When a part is not on hand, the logistics coordinator of the depot places an inquiry into the system and into other depots to find one. If the part can't be found close by, the depot sends an emergency order to the Logistics Department via remote communications, ARCLINK, DSSLAVE, and multiplexed lines. When the emergency network is used, the entire scenario takes less than 5 minutes. The necessary parts are immediately shipped, avoiding a potentially disasterous problem.

Just Imagine. . .

Let's take a look at what happens to an order — from the inside.

All orders are called into the Customer Support Center where they are entered into the system. These orders are placed using an 800 number: a sales coordinator takes your call and enters your order, as you speak, onto one of many terminals at the Center. These terminals are connected to the network through a multiplexed link, and the orders are immediately sent from the Customer Support Center to the Logistics Department for processing. That's where the real show begins.

If you had placed an order for a consumable that is stocked in the Logistics warehouse (like printer ribbons or disk packs), your order is printed on site, filled, and shipped immediately. Carrier and waybill information is also stored in another terminal as backup for customer inquiries. Orders for software or documentation are sent to Software Services via a DATAPOLL link and are filled from that point. If you've ordered something that isn't stored in San Antonio, such as paper, the order is sent to the Datapoint vendor for that product. This is what happens with paper.

Suppose you place an order for 10 boxes of paper for your 9621 160 CPS printer. Paper orders are "split" from the other orders received from the Customer Support Center. A DATABUS program "reads" the orders that arrive from the Center and builds a special communications file for them. At 9 a.m. and again at 3 p.m., a word processing operator in Logistics begins a program that opens IEOS, prepares a message, and appends a cover letter to the communications file that contains the orders for paper. The entire message is sent to a remote paper vendor in Dayton, Ohio, via the Electronic Message Service network controller and telephone lines. This vendor receives the information on a Western Union Teletype and fills the order from the remote location. Within 48 hours from the time you placed your order, your paper is on its way to you.

For more information concerning ARC local networks or remote communcations, please contact your local Datapoint representative.

ARCNET: Design and Implementation of a Local Network

This article was published in Electronics News, *and it was written by John A. Murphy from the Research and Development Department here at Datapoint Corporation. It is published in SOURCE DATA in its entirety.*

Introduced in 1977, Datapoint Corporation's ARC (Attached Resource Computer) local network is by far the most widely used local network on the market. The latest count shows that there are more than 4,000 ARC systems at customer sites in the United States, and another 830 installed in foreign countries.

The ARC system is based on Datapoint's ARCNET networking protocol, a multiple processor system designed for efficient resource sharing and modular expansion. The ARCNET protocol allows up to 255 Datapoint processors to share the same disk files, printers, and other peripherals at a communication rate of 2.5 million bits per second. Originally, the workings of ARCNET were considered proprietary, and only Datapoint processors could participate in an ARC network. Recently, however, arrangements have been made for other vendors' processors to use ARCNET, and the decision has been made to release the details of ARCNET to the public.

This article constitutes Datapoint's first public explanation of how ARCNET works.

The Motivation For ARCNET

In 1976 Datapoint started work on a communications system that would allow the efficient sharing of fast resources via a Resource Interface Module, or RIM, network interface. Design goals included:

- HIGH SPEED-a data rate in the megabaud range, on the same order of magnitude as the transfer rate of a disk,
- BUS OPERATION-the ability for any "node" (connected processor) in the system to send data to any or all of the other nodes,
- ADDRESSABILITY-the ability to send a message from one node to another without incurring any overhead at nodes not directly concerned with the message,
- SECURITY-the inability of any node to "eavesdrop" on messages not intended for it,
- DISPERSED CONTROL-the absence of any single master node or control point,
- FAIRNESS-the inability of any node to "hog" the system, either by sending long messages or by sending more than its share of messages,
- DYNAMIC RECONFIGURABILITY-the ability to add or delete processors by connecting them to or disconnecting them from the system, or by turning power on or off at any time, without disrupting the system,
- and, of course, the usual goals of infinite reliability and zero cost.

Development Of The ARCNET Protocol

The first breadboard RIMs used what has become known as a Carrier Sense Multiple Access (CSMA) scheme: a unit with traffic to send waited for the system to go quiet, waited for an additional random amount of time, and then started transmitting. It worked, but none of the people involved felt comfortable with the idea of intentionally sticking random variables into a system!

The next step was a contention-free system based on a self-polling scheme, or what is known today as token passing. Each unit had a unique address. At any given moment, one and only one unit could be considered the "master," or "in control" of the system. The "master," if it had traffic to send, did so; otherwise, it passed "control" to a new "master." An attempt to pass control was made by sending a token to the unit with the next sequential address. If the current "master" heard any activity at all, either the start of a data packet, or the continuation of the polling, or even an attempt by a new node to join the system, it relinquished control. If it heard no activity for the maximum amount of time required for the signal to make a round trip through the system, the "master" repeated the attempt to pass control using the next higher address.

For those who like to make comparisons with the various theoretical networking schemes, ARCNET at this stage of its development had characteristics very similar to BRAM, the Broadcast Recognizing Access Method described by Chlamtac, Franta, and Levin in 1979. There were, however, several significant differences that distinguished ARCNET:

> BRAM handles the passing of control, or scheduling, in a passive manner, with each node counting off a time interval from the last traffic;
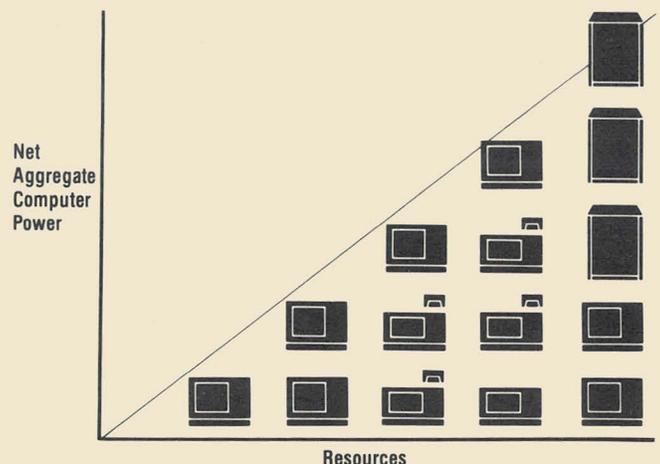
In an ARC system, the net available computer "horsepower" increases with each physical unit that is added.

ARCNET uses an active scheme, with each node being polled. In a physically large, high-speed system, where the time needed to pass the token from one node to the next is much less than the round-trip propagation time of the system, ARCNET spends proportionately less time passing control from node to node: BRAM takes an interval of time equal to the difference in node addresses, modulo the maximum number of nodes in the system, multiplied by the maximum round-trip propagation delay of the system. ARCNET takes only the time to send a short token message to the next active node.

ARCNET places considerably less severe accuracy and stability requirements on the clocks in the nodes: BRAM clocks must stay in sync for periods on the order of the maximum propagation delay multiplied by the maximum number of nodes, while ARCNET clocks are reset at least once every maximum propagation delay.

Under conditions of light loading, BRAM goes for relatively long periods with nothing at all being transmitted by any node; ARCNET exhibits a very regular and predictable pattern of transmissions, even with no traffic. While this may sound like a dubious advantage, it has important implications with respect to automatic reconfiguration of the system when the token is lost, when a new node enters the network.

ARCNET evolved into its final form with the addition of several optimizations. The polling method was modified so that each node remembers the address of the node to which it last passed control. Thus a dispersed list is kept of all the active nodes in the system, one entry being kept at each node. This greatly speeds the polling, or token passing operation, since no time is spent waiting for responses from non-existent nodes. A hardware packet acknowledgement was added to the protocol to reduce the number of packet exchanges, as well as the amount of time required for a higher level transaction.

The final protocol optimization was the addition of a FREE BUFFER ENQUIRY preceding the transmission of a packet. The motivation here was that ARCNET was designed to ease resource sharing: most of the traffic is between a large number of users and a small number of servers. It would be a waste of time to send a packet to a node that has just received one packet and has not had time to allocate buffer space for another. The FREE BUFFER ENQUIRY alleviates this problem by automatically asking the intended receiver if he has room for the packet before sending it; if the answer is NO, the token is passed on and the node tries again the next time it gets control.

ARCNET, like BRAM, has the nice property that the ratio of overhead to throughput decreases as traffic on the network increases. The time required to pass a token remains constant for each node, whether the system is working at zero utilization (the only traffic is the token) or 100 percent utilization (all nodes are transmitting after getting the token.) Thus, the overall efficiency of the system does not rise to a peak and then fall off, as in CSMA networks.

Another important property of ARCNET is that the time a node must wait for a chance to send a packet (the time it takes the token to get all the way around the network) is bounded. After any node sends a packet, all the nodes in the system get a crack at the bus before the first node gets another chance.
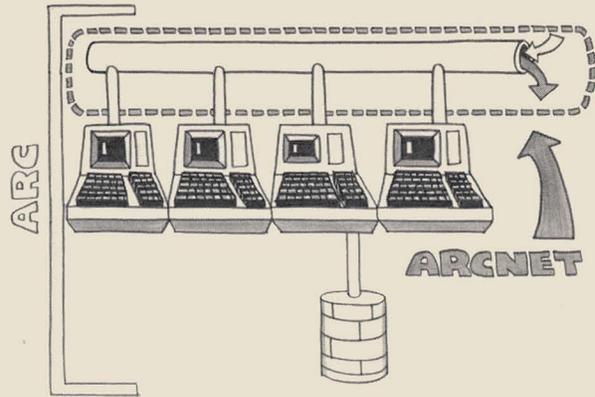
ARC and ARCNET are distinctly separate yet functionally interdependent.

Transmission Media and Network Topology

The ARCNET protocol imposes very few restrictions on the transmission medium. Basically, any node must be able to hear any other node's transmissions, there must be only a single path between any two nodes, and signals must be able to pass between any two nodes in no more than 31 microseconds (us). Thus an ARCNET system may be implemented using any of a number of communications facilities including both broadband and baseband coaxial cable systems.

In a broadband cable system, the transmission medium (the cable itself, its associated amplifiers) is shared among a number of communications services. Several totally independent ARCNET systems could share the cable with a number of low-speed terminal-to-computer links or various video services, for example. All communications paths pass through a central point, or headend. Each user, such as an ARCNET node, transmits on one frequency and receives on another. The headend equipment receives on the user's transmit frequency and "repeats" whatever it hears onto the user's receive frequency.

The advantage of a broadband system is that it provides for the distribution of a number of different services on the same cable. The major disadvantages of a broadband system are complexity and high cost. The transmitters and receivers required at the nodes, the line amplifiers required in a system of any size, and the headend equipment are all relatively expensive. Also, the planning and installation of the broadband cable itself can be a little tricky. All of the taps that will ever be required in a given run of cable must be present when the system is first installed.

Baseband

The advantage of a baseband system is simplicity and low cost. Amplifiers in a contention-free system such as ARCNET don't have to be "bi-directional" since only a single node is transmitting at a time, in one direction. Actually, the "amplifiers" on a baseband system are

generally signal re-generators. Baseband amplifiers "know" that the signal stream will be composed of data bits instead of cable TV pictures (or piped music), so they can be designed to improve the quality of the signals passing through them. Ordinary amplifiers inevitably degrade the signal.

The disadvantage of a baseband system is the difficulty of tapping the line for distribution. One of the more unpleasant realities of real world communications systems is the existence of reflections. Reflections are the echos that "bounce back" when a signal hits an impedance mismatch in a cable. An impedance mismatch is the electrical "rough edge" that exists anywhere a tap is connected to the cable. And since the reflections from a number of different taps can accumulate and "gang up" on the real signal, the quality of the taps can quickly become the limiting factor in the number of nodes allowed on the bus.

The solution used in broadband systems is a device called a directional coupler, which relies on the fact that all broadband nodes communicate with each other only through the headend—the nodes transmit to the headend on a low frequency, and the headend rebroadcasts the signal on a high frequency. The directional coupler forms a tap which discriminates against anything coming from downstream (the direction away from the headend). In a broadband system that means reflections; in a baseband system that means reflections and signals!
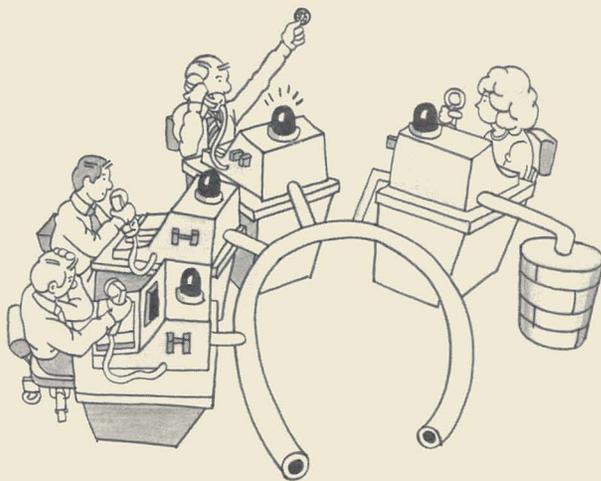
On the ARCNET "party-line" those who want to "talk" have their "phone off the hook." But at any given time, the only user who can use the line is the one with the token.

The ARCNET Solution

The standard ARCNET interface is to a baseband system using RG62 coax. A unique coding scheme is used; a 200 nanoseconds (ns) wide dipulse (a 100 ns wide positive pulse immediately followed by an identically shaped negative pulse) is transmitted for a mark and no energy at all is transmitted for a space. This results in a system with virtually no intersymbol interference. The use of transformer coupling and matched filters adds to the reliability of the scheme.

The reflection problem, as well as some fault isolation and routing problems, is circumvented by implementing the network as a free-form tree, or collection of stars. Each node connects, through a length of coax, to a port on an

ARCNET "hub"; the coax is properly terminated at each end and no taps of any kind are used. In a sense, the hub may be thought of as an amplifier with a number of ideal taps, all mounted in the same box!

These "taps" are ideal in the sense that they have no insertion loss and no tap loss, and yet provide total suppression of reflections, even from unterminated lines. Each of the ports of a hub may be connected to an ARCNET node, to another hub, to an unterminated length of coax, or to nothing at all.

While centralizing the location of the cable taps in a dispersed computing system may seem counter-productive, there are a number of practical reasons for doing so. Both initial and add-on cabling of a site are simplified, since cables may be fanned out from the centralized tap locations (typically utility closets) without having to consider any critical spacings, and the only things that have to be installed inside walls or ceilings are the cables themselves. Extra cables may be put in for future expansion, or for equipment mobility, without the cost of unnecessary taps. Cables may be routed through the same raceways and conduits used for the telephone system, which has the same sort of topology.

Numbers, Numbers, Numbers

Having dealt with some of the historical aspects of ARCNET, we will now take a detailed look at the operation of the network. ARCNET is a serial, asynchronous system utilizing 11-unit start-stop operation at 2.5 megabaud. Like most communications systems, ARCNET imposes a limit on the maximum propagation delay through the system; no two nodes may be "farther" than 31 microseconds (us) apart.

Translating the 31 us into a physical distance limitation requires some knowledge of the transmission medium and the characteristics of any intervening repeaters or similar devices, but something on the order of four miles is easily obtained.

The bus idles in a spacing (no signal) condition. Each transmission starts with an alert burst consisting of six unit-intervals of mark (signal). Eight-bit characters are then sent with each character preceded by two unit-intervals of mark and one unit-interval of space. There are five different types of transmission:

- INVITATIONS TO TRANSMIT-An ALERT BURST followed by three characters; an EOT (End Of Transmission) and two (repeated) DID (Destination IDentification) characters. Used to pass the "token," or control of the line, from one node to another.

- FREE BUFFER ENQUIRIES-An ALERT BURST followed by three characters; an ENQ (ENQuiry) and two (repeated) DID characters. Used to ask a node if it is able to accept a packet.

- PACKETS-An ALERT BURST followed by from 8 to 260 characters; an SOH (Start Of Header), an SID (Source IDentification), two (repeated) DIDs, a COUNT, from 1 to 253 data characters, and two CRC (Cyclic Redundancy Check) characters. Used to move data between nodes.

- ACKNOWLEDGEMENTS-An ALERT BURST followed by one character; an ACK (ACKnowledgement). Used to acknowledge PACKETS and as an affirmative response to FREE BUFFER ENQUIRIES.

- NEGATIVE ACKNOWLEDGEMENTS-An ALERT BURST followed by one character; a NAK (Negative AcKnowledgement). Used as a negative response to FREE BUFFER ENQUIRIES.

The receiver validates all incoming transmissions by checking for:

- At least one mark and exactly one space preceding each character
- An EOT, ENQ, SOH, ACK, or NAK following the ALERT BURST
- Proper number of characters (3, 8 to 260, or 1) characters (3, 8 to 260, or 1)
- At least nine spaces following the last character.

Each node in a system has a unique ID (IDentification) from 1 to 255. ID 0 may not be assigned to any node since destination 0 is used to indicate a BROADCAST to all nodes. System operation is based on a token being passed around the system with each node passing it to NID (Next ID), the node with the next higher ID in the system. When a node receives an INVITATION TO TRANSMIT containing its ID, it "receives the token" and assumes control of the line. The ID of the next node is determined during SYSTEM RECONFIGURATION.

A SYSTEM RECONFIGURATION occurs any time the system determines that there is no token circulating. Under normal conditions there are never more than 78 us between transmissions. When any node sees that the line has been idle for 78 us it knows the system is being reconfigured and initializes its NID to its own ID. It then starts a time-out equal to 146 us, times the quantity 255 minus its own ID. If this time-out expires with no line activity, the node starts sending INVITATIONs TO TRANSMIT. (This time-out expires first for the node with the highest ID in the system.)

After sending an INVITATION TO TRANSMIT, the node waits for the receiving node to respond with a FREE BUFFER ENQUIRY, PACKET, or INVITATION TO TRANSMIT. If there is no activity for 74 us, the node increments NID and tries again. If it hears any activity before the time-out expires it releases control of the line. During SYSTEM RECONFIGURATION, INVITATIONs TO TRANSMIT will be sent to all possible IDs. Each node, however, will have saved NID, the ID of the node that assumed control from it.

From then until the next SYSTEM RECONFIGURATION (which will occur only when a new node is powered up, or when a node gets dropped from the system due to line errors causing it to miss the token) control is passed directly from node to node, with no time wasted sending INVITATIONs TO TRANSMIT to IDs of nodes that have been turned off or disconnected.

The time required to do a SYSTEM RECONFIGURA-RECONFIGURATION is a function of the number of nodes in the system and the highest ID among them, and falls between 24 and 61 ms.

Adding A Node

Once the mechanism for recovering from a lost token exists, the obvious way for a new node to enter the system is to destroy the token! When a node is first powered on or has not received the token for approximately 840 ms, it sends a reconfigure burst consisting of eight marks and one space repeated 765 times. The purpose of this burst is to destroy the token and terminate all activity on the line. It is longer than any other type of transmission and will therefore interfere with the next INVITATION TO TRANSMIT and keep any node from seeing it, "getting the token," and assuming control of the line. It also provides line activity so that the node sending the INVITATION TO TRANSMIT releases control of the line. Thus, the node that had control releases it, and no other node picks it up; the token has been destroyed, resulting in a system reconfiguration.
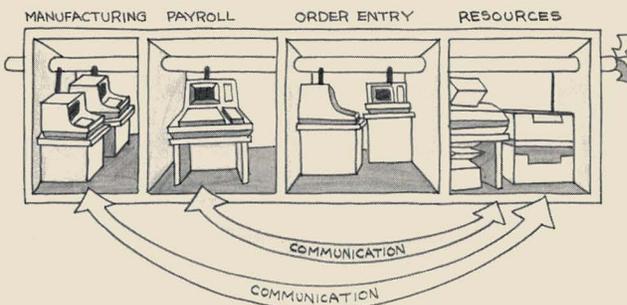
Since ARCNET has its own intelligence built in to handle communications between the elements in the system, the processors can devote themselves to productive work.

A system reconfiguration is not required when a node is turned off or disconnected from the system. The node that had been passing the token to the absent node hears no response to its INVITATION TO TRANSMIT, increments NID, and sends another INVITATION TO TRANSMIT. This sequence continues until an INVITATION TO TRANSMIT is sent to the node that was receiving the token from the removed node.

Node Operations

An ARCNET node is physically integral to a processor and is logically connected to the network through a status register, a command register, and some number of 256-byte memory buffers. The processor is unaware of any of the details of the token passing operation and need only concern itself with transmitting and receiving messages.

To transmit a message, the processor writes the destination node ID, the length of the packet, and the packet data into one of the buffers and gives a command to transmit that buffer. (By convention, the first byte of the packet data is a SYSTEM CODE, allowing multiple systems with entirely independent packet formats to coexist on the same network.)

The status register contains two bits related to the transmit operation: TA (Transmitter Available) and TMA (Transmitted Message Acknowledged). TA goes false when the transmit command is given and comes true again when the node has finished the transmit sequence described below, at which time TMA indicates if the destination node acknowledged the packet.

To receive a message the processor issues a command to enable reception (optionally including reception of broadcasts) into a particular buffer. Status bit RI (Receiver Inhibited) goes false when this command is given and does not come true again until after a packet has been received.

When a node receives the token it checks to see if it has a packet to send, i.e. if TA (Transmitter Available) is false. If not it passes the token to NID. Otherwise it tests the DID (Destination IDentifier) in the transmit buffer. If this byte is 000, the packet is a BROADCAST and the node sends the packet without further action. Otherwise it sends a FREE BUFFER ENQUIRY to the DID node and waits up to 74 us for a response. If the response to the FREE BUFFER ENQUIRY is an ACK, it sends the packet. If the response to the FREE BUFFER ENQUIRY is a NAK, it passes the token to NID and will send another FREE BUFFER ENQUIRY the next time it receives the token. If the node times-out waiting for a response to the FREE BUFFER ENQUIRY it sets TA and passes the token to NID.

After sending a packet the node waits up to 74 us for a response. If it receives an ACK it sets TMA and then TA and passes the token to NID. If it times out waiting for an ACK (packets are never NAK'ed) it just sets TA and passes the token to NID.

When a node receives a FREE BUFFER ENQUIRY it tests RI. If RI is true it sends a NAK. Otherwise it sends an ACK.

When a node receives an SOH (indicating the start of a PACKET) it writes the SID into the receive buffer and then checks the first DID. If this byte is 000 (indicating a BROADCAST) the node tests to see if reception of BROADCASTs is enabled. If reception of BROADCASTs is enabled, or if the first DID is the node's own ID, the node writes the second DID and the rest of the PACKET into the receive buffer. Otherwise, it ignores the rest of the PACKET.

If after being written into the receive buffer the PACKET fails either the CRC or length validation phases, the node ignores it. Otherwise it tests the DID in the receive buffer. If the DID is 000 for BROADCAST, the node simply sets RI. If this byte is the node's own ID, the node sends an ACK before setting RI. If the DID is neither 000 nor the node's own ID, the node ignores the packet.

Performance Factors

The most interesting aspect of performance in a local network is usually the amount of time a node may have to wait before being able to send a message. In a token passing scheme, this waiting time is bounded by the time it takes the token to make the rounds of each and every node. There are several possible sequences of events that can occur when a node receives the token; two of these make up the vast majority of cases, the others involve error conditions, messages sent to non-existent nodes, and the like.

The two cases of interest are a simple token pass and a message followed by a token pass. A simple token pass takes about 28 us (ignoring propagation delay). A message followed by a token pass takes about 141 us plus 4.4 us per byte of data (again ignoring propagation delays). Thus the time required for the token to make a complete trip around the network is approximately 28 us per node plus 113 us per message plus 4.4 us per byte.

In a system, for example, consisting of 10 nodes and sending messages no longer than 100 bytes, a complete token trip will be at least 280 us (when no messages are sent) and no longer than 5,810 us (when all 10 nodes send 100 byte messages). If only a single node is sending messages, it can send one every 833 us: a rate of 1200 messages, or 120,000 bytes, per second. If all 10 nodes are sending messages, each one can send one every 5810 us, a rate of 172 messages per second per node.

At the other end of the size spectrum, the original ARCNET system in the Datapoint Research and Development Department now has anywhere from 150 to more than 200 nodes active at any given time. It supports two totally independent operating systems and a wide variety of uses including program loading, word processing, print spooling, program development, electronic mail. The traffic load rarely falls below 400 messages per second, yet less than 2% of the nodes send messages on the average token trip. The time required for a token trip, therefore, stays very close to the no-traffic value; peaks of three times the no-traffic value are extremely rare.

Conclusion

In the five years that Datapoint has been shipping ARCNET systems, more than 20,000 nodes have been installed. ARCNET is currently in worldwide use at thousands of customer sites. The RIM hardware has been implemented in a number of different forms, ranging from the original 900 cu. in. shoebox-sized model using MSI TTL, to the current 35 sq. in. boards designed around custom MOS LSI chips. Computers and peripherals are available which have an ARCNET coax connector as their only I/O port. The use of the ARCNET protocol, and even of the LSI chips that implement it, is being made available to all interested parties.

If it is true that the real industry standards are set not in the literature, but in the marketplace, then ARCNET may well be considered the existing standard for local area networks.

BRAM: The Broadcast Recognizing Access Method
Imrich Chlamtac, William R. Franta, K. Dan Levin
IEEE Transactions on Communications
Vol. COM-27 No. 8. August 1979, pp 1183-1189

Performance Issues in Back-End Storage Networks
Imrich Chlamtac, W.R. Franta, P.C. Patton,
Bill Wells, Computer, February 1980, pp 18-31

Custom Chip Handles Local Net Interface
Electronics, September 22,1981, p 41

'Old' Local Net Wins Big New Backer
Electronics, September 22, 1981, p39

Datapoint May Go Public With its Chip for Local Networks
Electronics, March 10, 1982, p 33

Concepts of RMS: RMS as an Operating System

This is the first in a series of articles concerning concepts of RMS. We're starting from the basics: in this article, we'll discuss RMS as one available operating system.

RMS, the Resource Management System, is a true operating system. It is responsible for the normal operating system functions such as physical I/O handling, memory management, "time-sharing" management and, more importantly, the multitasking and resource sharing capabilities. To manage all these functions, the RMS Nucleus, RMS' active management program, is resident in every node. (By the way, in RMS terminology a processor is called a "node".) The nucleus takes up some memory and processing power while fulfilling its management role. The amount of memory and node overhead consumed by RMS depends on the configuration of the specific node—depending on the number of resources that must be managed, the number of tasks that may be run, and the way in which networking is being supported.

Exactly What Can RMS Provide?

A primary capability of RMS is that of multitasking. The ability to multi task allows many users to share the power of the node while doing many different things. Data processing programs written in languages like COBOL, DATABUS, and RPG can be run on the same processor at the same time as the word processing program IEOS is being run. These tasks are started by users through the keyboard and screen of workstations. A workstation is simply a device with a keyboard and a screen that allows the user to interact with the RMS operating system. The Datapoint 8200 and 8220 terminals along with the console of the Datapoint 5500, 6600, 6000, 3800, and 8600 processors can be used as workstation devices. The number of tasks that can be run on any processor is limited by the memory available and the speed with which the user wishes to process information.

RMS can also provide resource sharing. During the multitasking function described above, RMS allows the users to access the various peripheral devices located on their nodes or, in a network, other nodes. Since each resource, whether it is a printer, disk, communications device, magnetic tape, card reader, node or network, is provided with a unique name, each node is an individually addressable resource. For instance, if there were two printers on a node, the first printer being named "P1" and the second being named "P2", a user could choose to use printer "P1" while a second user is using "P2". The resources need not be directly connected to the node where the user's workstation is located. Through the networking of the Attached Resource Computer (ARC), the higher cost resources such as disk drives and printers can be shared by users attached to any node in the system.

The RMS implementation of data communications provides the user with flexibility not found in other computer systems. The communications device is considered a named resource. Most systems require that the user be attached to the processor where the communications device is located. With RMS, the communications device can be located anywhere on the network and can be addressed by its own unique name. Now the communications interfaces can be kept in the basement in a locked room, though the user on the 5th floor can access them without moving from his or her chair.

Access to Information

The next question is probably, "How do I get access to all of these resources?" In RMS, each user is allowed to set up an environment. This environment points to the resources that the user needs to access. In essence, the environment is the users view of the computer system, be it stand-alone or spread throughout a large ARC network. The environment can also be changed at will during the on-line session by using the ENV or environment command. When the user logs off of the system, the environment definition is removed from the active system; it is established again when the user logs on.

"Can I protect my resource from outsiders or unauthorized access?" In one word, yes. RMS provides some of the most complete security protection found in the computer industry today. Part of the process of establishing an environment is the entering of passwords. Each single file resource such as a printer, card reader, or communications device can be protected by a password so that only users with the correct password can gain access to them. Multiple file resources (disk files) may also be password protected to insure the security of the data. A part of the log on process may also require a password if it is required by the system manager.

RMS provides many advantages for the system designer, programmer, and, most importantly, the user. As we continue in this series of articles about RMS, we will provide you with tips and hints to make RMS easier for you. For more information on the topics we've discussed here, refer to the RMS User's Guide Volume I and to the RMS User's Guide Volume II, the ENV Utility.
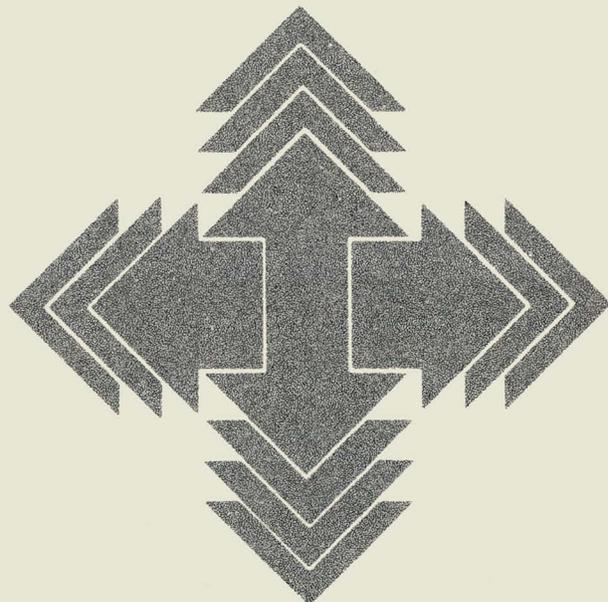
DATABUS and BASE 95

The June/July issue of *Source Data* (Issue #19) included an article that outlined some quirks regarding round-off and truncation of integer values in DATABUS. In that article, the author described one of the uses for the discoveries as the capability to convert base 10 numbers to base 34 so that they could be stored in less space. At the end of the article, the author stated that the ideal maximum storage density, at least in the DATABUS language, would be to use base 96 but that the computer cycles to achieve this would be prohibitive. In this article, I will outline a method for achieving this result that requires no more effort than a base 34 conversion and, in fact, includes very minor overhead.

ENCODE/DECODE

In the pharmacy system that we have written, we developed an Encode and a Decode routine that will convert single digit numbers in the range 0 to 94 into a single alphanumeric character in the range of 40 to 176 octal. Encode converts the number into the alpha, and Decode converts it back. Using the techniques outlined in the earlier article, we frequently use this technique to convert multi-digit numbers into base 95 so they may be stored in half their standard area. In the layout below, we are using base 95 and not base 96. Base 96 would be achievable with no extra effort; however, we have encountered the need to reserve the DEL character (0177) as a special flag (such as to designate a -1).

The heart of the conversion is a simple 95-byte DIM field that holds, in order, the ASCII string of characters ranging from a space to the character just prior to the DEL (~). This yields 95 printable, storable characters that can be used to hold the numbers ranging from 0 to 94. Both the Encode and Decode routines outlined below consist of only 4 instructions and neither involves a loop. The Decode routine does require the advanced DATABUS instructions available in all versions of DATABUS except the lowest implementation on the 1550. DS1500 does have these instructions. A different version of Decode could be implemented on the single terminal version of DATABUS (DB15), but it would require looping. We used such a version initially with success.

Assume that the 95-byte ASCII string is named ASCIICDS (for ASCII codes). Encode is called with the source number in a 2-byte FORM (NWORK2), and the result is returned in a 1-byte DIM (WORK1). Decode operates in the reverse order. The code for Encode and Decode is as follows:

```
ENCODE  ADD "1" TO NOWORK2          •account for a base of 0
        RESET ASCIICDS TO NWORK2    •set the FP to the encode result
        MOVE ASCIICDS TO WORK1      •translation result
        RETURN
DECODE  RESET ASCIICDS TO WORK1     •set FP to ASCII value-37 octal
        MOVEFPTR ASCIICDS TO NWORK2 •determine the offset
        SUB "1" FROM NWORK2         •account for a base of 0
        RETURN
```

This approach also lends itself to the writing of very simple extension routines that increments and decrements the encoded number without decoding it first. NXCODE increments the encoded number in WORK1, and PRCODE decrements it.

```
NXCODE  RESET ASCIICDS TO WORK1
        BUMP ASCIICDS BY -1
        MOVE ASCIICDS TO WORK1
        RETURN
```

Why Bother?

When a system designer must place a large database into a small area, it can be very important to save every byte. Or when certain system parameters are used optionally throughout the system, it can be most beneficial to reduce their size, especially if they are not needed for arithmetic. In our pharmacy system, we must write the terminal number in the pharmacy system but did not wish to be limited to 9 terminals or use two bytes. Instead we encode the terminal number into one byte giving room for up to 94 terminals.

Another frequent use is in the date fields. Most everyone stores the date in the form YYMMDD. This requires 6 characters and does not account for people born in the last century or the eventual problems that will be encountered in the year 2000 (when 90% of all computers will crash). We have used this encoding technique to allow us to devise a date format that requires only 5 characters but that will handle all dates from the year 1800 to 2799 (by which time I will be retired). The format is YYYMD where YYY is a 3-digit year that ranges from 000 to 999, M is an encoded month, and D is the encoded day. In less space, we are storing more information, and in a system that uses as many dates as a pharmacy system does, this is extremely helpful. This date format is not all cumbersome, since the routines to encode are very swift and dates are most often used for comparisons and not arithmetic. If arithmetic is required, Decode can always be implemented swiftly.

The remaining most frequent use for this encoding technique is for the storage of archival dollar data that is seldom accessed but is required once a year (or for sales demonstrations). For instance, a dollar value as large as $8,573.74 may be stored in only 3 bytes using this base 95 compression.

Let's not forget the numerous little system counters that always exceed 9 so they cannot be placed in one byte, or they exceed 26 so that even the alphabet cannot be easily used to store their values.

Base 95 (or 96 if you prefer) can be easily implemented. I recommend it for any databases that can benefit from low cost streamlining.

Association Performs Benchmarks on 8600

The Association of Computer Users, an independent, non-profit organization with headquarters in Boulder, Colorado, has performed benchmarks on a number of multi-user computer systems designed to provide "real life" comparative information. The latest Benchmark Report covers Datapoint's 8600. The evaluations have been published in the ACU's Benchmark Reports series. Articles derived from the reports have appeared in publications such as *Computerworld, Modern Office Procedures,* and *Interface Age.*

A group of Datapoint employees from San Antonio went to Boulder to help with the 8600 benchmarks. Back in San Antonio after the testing, we took the systems comparisons one step further. The same DATASHARE program used during the ACU test of the 8600 was used to conduct benchmarks on several other Datapoint systems.

This article compares the "real life" performance of several Datapoint DOS and RMS systems. Included in the results are timings of 8800, 8600, and 6600-based standalone systems.

The Benchmark Process

The objective of the benchmark was to measure response time degradation as terminals were added to the various systems. A Keyin Simulator program was used to simulate from one to 24 8200 terminal users interacting with the DATASHARE order entry program. This simulator was used to execute the DATASHARE program and measure system response times.

The order entry program is based on a system designed for functions that the ACU needs in its day-to-day business. The ACU must perform three types of processes for its membership: new membership entries, membership renewals, and payments.

When a new member joins the Association, information is input by the operator and stored in a member file. A packing slip is then prepared indicating the items to be shipped. For membership renewal, the old member record is retrieved and updated, and a packing slip is prepared. In both cases, if payment has not been included, an invoice is prepared and the billing information is stored in an accounts receivable file. When a payment is made, the receivables record is retrieved and updated.

The Results

The graph shows the results of the benchmarks. This "real life" order entry job operated in a DATASHARE-only environment running standalone under both DOS and RMS.

The 6600 DOS and 8800 systems had the fastest response times. The timings were close when these systems were configured with up to 6 terminals. However, with more than 6 terminals operating, the 8800 response times were significantly faster. In addition, with more than 6 terminals, the 8800 could support up to 75% more ports at a given response rate than the 6600 DOS system.

The response times for 8600 and 6600 RMS systems were close. These systems supported, at a given level of response, half as many terminals as the 8800. For this order entry task, the 6600 RMS system supported a limit of 14 terminals. (At the same response rate, the 6600 DOS system supported 18; at a greatly degraded response rate, it supported 24 terminals.)

Remember, these benchmarks were DATASHARE-only. In this environment, the 8800 is the choice for a large standalone DATASHARE system — it provides faster response and can support more terminals. For the same reasons, the 8800 also serves well as a file processor for a large data base in an ARC system. For applications which require word processing or multifunction capabilities, the 8600 offers excellent performance.

Configurations				
Model	Processor	Memory	Disk	
8860	8800	640K	202Mb	(1PP/MPCA)
8630	8600	256K	20Mb	
8620	8600	256K	10Mb	
4750	6600	256K	120Mb	DOS
4750	6600	256K	120Mb	RMS
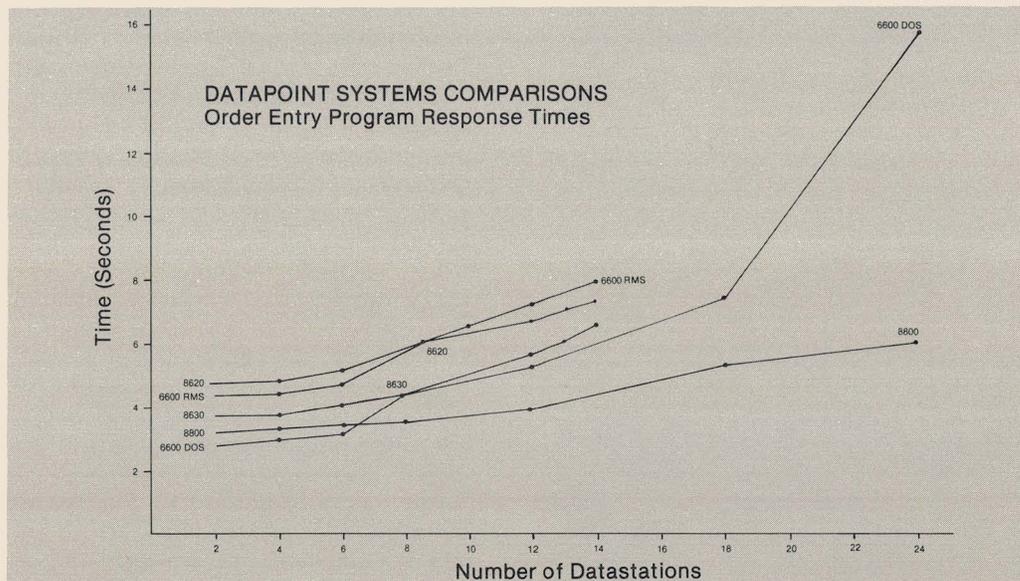
DATAPOINT SYSTEMS COMPARISONS
Order Entry Program Response Times

An All-Purpose CHAIN for Word Processing

Sitting at a workstation waiting to enter the next command to IEOS can take quite a bit of your time, especially when you have a number of documents to print or file, for example. By using a CHAIN file, you can start and finish several jobs while you're away from your workstation or processor.

The First Step

Create a document in IEOS to perform the CHAIN. You can name the document anything you'd like. In this example, we'll write a CHAIN to print documents from a library called MEMOS, file them in the ARCHIVE library, and then PACKUP the library MEMOS, deleting its AIM file. First, let's create the document.

CREATE DOCUMENT PRTFILE

When creating CHAIN files, you'll need to follow special rules for formats. Use the following format for your file:

```
JUSTIFY           LEFT
PAGE LENGTH       1
LINE SPACING      1
PARAGRAPH         1 0
SET TAB
LEFT HEADER
RIGHT HEADER
LEFT FOOTER
RIGHT FOOTER
PAGE NUMBER
TOP MARGIN        0
BOTTOM MARGIN     0
LEFT MARGIN       1
RIGHT MARGIN      80
PITCH             10
```

Once you've completed the format changes, you'll enter the following information to print and file documents:

```
IEOS
OPEN LIBRARY MEMOS
PRINT D DOCUMENT1
PRINT D DOCUMENT2
PRINT D DOCUMENT3
FILE D DOCUMENT1 AS D NEWDOC1
FILE D DOCUMENT2 AS D NEWDOC2
FILE D DOCUMENT3 AS D NEWDOC3
UTILITY
UNLOCK LIBRARY MEMOS
YES
IEOS
PACKUP NO LIBRARY MEMOS
EXIT
```

You can enter commands for as many documents as you want to be printed or filed. Just remember to use the exact command line that is needed to execute a command in IEOS. If there is just one mistake, the command will not be executed.

Once you've entered all the information you want and have ended the document, OUTPUT the file to a text file using the command line:

```
OUTPUT DOC PRTFILE ON TXT = PRTFILE/TXT    (in DOS)
OUTPUT DOC PRTFILE ON TEXT = PRTFILE/TEXT  (in RMS)
```

To execute the CHAIN file, exit from IEOS and enter the command line:

```
CHAIN PRTFILE/TXT   (in DOS)
CHAIN PRTFILE/TEXT  (in RMS)
```

and press the ENTER key. Sit back and watch your workstation do all your work for you!

You can use this CHAIN file to perform any repetitive task, like filing, printing, deleting, or outputting documents. Just be sure that you enter the command in proper IEOS syntax, so that you won't have problems with incomplete execution. Happy CHAINing!

Multiplan and the 1550

In September, Multiplan* 1.1 was released for use on 1550 machines with 96K of memory.

Multiplan, which lets a processor perform powerful numerical calculations in an accounting spreadsheet format, has been available for several months on the 1800, 3800, and 8600 processors (see *Source Data* #18, April/May 1982). The version now released for the 1550 is identical in its operation to that of the previous version.

The 1550 version of Multiplan will operate on any 1550-compatible media. These include single-sided dual-density, double-sided dual-density diskettes, and 10MB cartridge disks. Software may not be ordered on the 10MB cartridge; it is transferred from a diskette.

The following is the list of model codes relevant to Multiplan 1.1 on the 1550:

Model Code	Description	Cost
9854	Multiplan software	no charge or maintenance
20829	SSSD diskette media	$15.00 (1 diskette)
20793	DSDD diskette media	$15.00 (1 diskette)
50697	Simplified User's Guide	$10.00
61370	Reference Manual	$7.55
61300	Quick Reference Card	$0.30

For further information on this release of Multiplan, please contact your local Datapoint representative.

ACD Performance Planner
Measures Effectiveness

The ACD Performance Planner, a software package that helps operations managers plan the effective use of Datapoint's Automatic Call Distributor, is now available.

The ACD Performance Planner gives you the capability to evaluate an ACD telephone system under varying traffic loads, routing patterns, and agent staffing levels. The Planner uses a simulation technique, with a varying number of lines and operators, for instance, to act out system performance under parameters you can select, as the operations manager.

The resulting report on use of lines and staff, as well as the quality of service provided to incoming callers under the simulated conditions, can be used to help plan for future system requirements.

Its Capabilities

The ACD Performance Planner can simulate up to 2500 calls in a busy hour of traffic activity, dispersed across as many as "four" ACD systems, with each "ACD" containing up to 16 telephone line splits and 16 agent groups.

Simulation parameters can be obtained from performance reports generated by the ACD. The operations manager can specify or modify the system traffic, split, agent, and routing assumptions through the menus associated with the Performance Planner. By changing the simulation parameters and executing the Planner again, a manager can evaluate alternate system configurations.

One Example of Its Use

One company's advertising department might plan to launch a direct response television campaign, with 1,000 incoming calls an hour expected during peak viewing hours. Past performance reports from the ACD indicate that most callers remain on hold for 35 seconds before being connected to an operator. The advertising department wants to make sure that no caller is kept waiting more than 15 seconds; therefore, they would need to know how many additional operators and lines are required to handle the projected calls. By using the Performance Planner, the manager can set up the proper configuration to make sure these requirements are met before the problem arises.

Performance Planner Reports

The ACD Performance Planner generates three reports that assist the operations manager in determining agent staffing requirements and the call routing steps that will optimize the agent workload and efficiency. The three reports are the Simulated Call Performance report, the Call Distributor report, and the Simulated Interflow Performance report.

The Simulated Call Performance report is generated for each split specified in the simulation. The simulated and user-specified parameters are reported in four sections: the current traffic level, simulated call distribution, simulated call performance, and simulated split traffic.

The Call Distribution report, generated for each ACD in the simulation, gives the percentage of calls handled by each agent group for each split in the ACD.

The Simulated Interflow Performance report, generated for simulations that include multiple ACD systems configured with interflow lines, reports how calls are switched from one ACD to another.

For further information concerning the ACD Performance Planner, please contact your local Datapoint representative.

Datapoint's CDR Enhanced in New Release

In November, Datapoint announced the release of Version 2.1 of the Call Detail Recorder (CDR). The CDR provides an affordable way to record details on telephone calls placed through a company's private branch exchange or key telephone system. The CDR system will calculate call cost, assign charges to the caller, and provide management reports on call volume, expenses, and use of telephone facilities. An explanation of the enhancements follow.

Reporting Enhancements

CDR version 2.1 allows reporting capabilities at the extension level so that reporting can be performed on incoming calls. This is important for users with service or operation attendant applications with high volume incoming calls.

Line use reporting capability will be provided to allow reporting on the actual line used where this information is supplied by the PBX. This gives the user greater control over existing telephone facilities. Local call reporting can be expanded to maintain a total number of local calls or a local call summary.

More comphrensive reporting will be added to the Cost Center, Department, Division, and Corporate reports. This will include a breakdown of local, long distance, and incoming calls available for this reporting structure.

System Operation Enhancements

Call buffering will be provided to ensure that no call records are lost during periods of high activity. This includes the capability of reading call records from a disk file.

The ability to handle account code records from the Northern Telecom SL-1 PBX will be added. The CDR has always had account code reporting capabilities for PBXs providing single call records per call, unlike the SL-1 which provides separate account code records.

Screen formats will be improved to provide more meaningful groups of information in an easier to read screen display.

If you'd like more information on the CDR, please contact your local Datapoint representative.
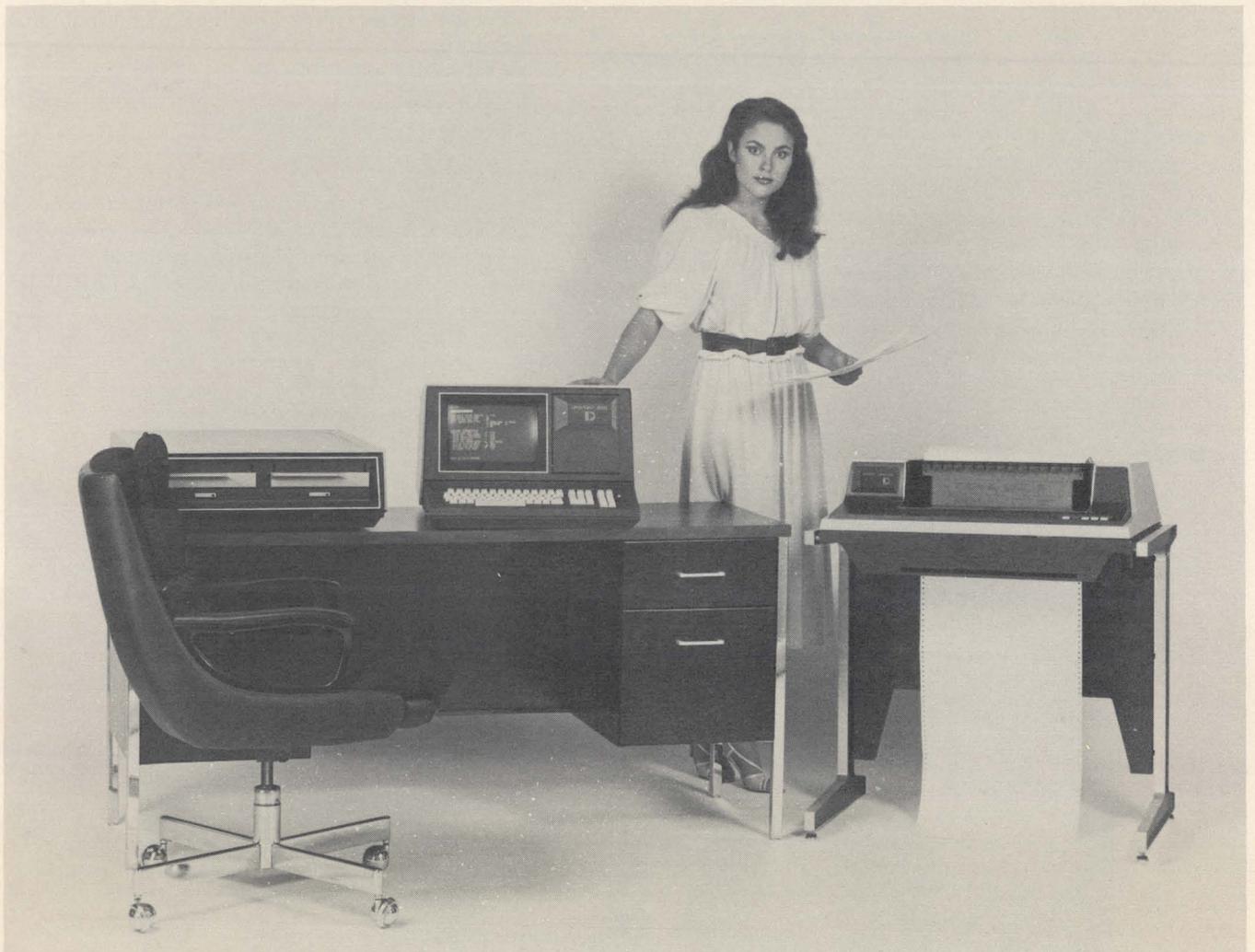
Datapoint CDR

Simplified Users Guides
for DATABUS

Datapoint processors are scattered throughout the world, meeting the needs of thousands of users. Each of these applications is unique, yet one key element can tie them together—the DATABUS language. DATABUS is one of a variety of computing languages, including RPGPLUS, COBOL, and BASICPLUS, available for use on Datapoint systems. DATABUS is a total business programming language that lets you fully utilize all of the capabilities of your particular Datapoint computing system. Whether you are using an 1800, 6600, or 8800 processor, or even an Attached Resource Computer (ARC) system applications processor, DATABUS is the computing language that can solve many of your computing problems.

Because DATABUS is a high-level language, many of the detailed programming aspects of computer usage are taken care of by the DATABUS compiler, leaving you to code your programming application in simple, English-like statements. DATABUS enjoys a special status in the Datapoint family of computer languages because it constitutes a powerful and sophisticated tool in the hands of a professional programmer and at the same time permits easy mastery and use by the non-professional programmer, such as a business executive, for applications essential to his job duties.

DATABUS is not only one of several languages available for single-user computing applications, but it also is the computing language used in all DATASHARE applications. DATASHARE is Datapoint's business timesharing system that enbles up to 24 users to simultaneously access the computing capabilities of a single processor.

What you've just read is the forward to Volume One of the RMS and DOS *Concepts of DATABUS Programming Simplified Users Guides.* These guides give you information on all aspects of programming in DATABUS, from simple explanations about the language to complex considerations of its intricacies.

DOS Guides

The simplified users guides for DOS come in three volumes. The first volume (Document No. 50004) explains:

* Flow Charting
* Displaying on the Screen
* Using the Keyboard for Data Entry
* Printing the Data
* Arithmetic
* Loops
* Disk Operations
* Using Subroutines
* Compiling Programs

Volume Two (Document No. 50516) is an advanced DATABUS book that takes you through typical business programming applications, explaining advanced language features as they are introduced. It details:

* System Analysis
* Designing the Data Base
* Data Entry
* Review of Indexed Access Methods
* Data File Manipulations
* Character String Manipulations
* Report Generation

Volume Three (Document No. 50049) is the most authoritative simplified guide available for advanced work in DATABUS. The third volume explains:

* Effective Data File Use
* System Security
* Virtual Memory Programming Considerations
* Printing
* ROLLOUT, SHUTDOWN, and CHAIN
* DSSLAVE
* Increasing System Performance

RMS Guides

The simplified users guides for RMS DATABUS follow those for DOS DATABUS closely, taking into account, of course, the changes made in the language for the different operating systems. The guides for RMS DATABUS are also available in three volumes.

Volume One (Document No. 61139), Volume Two (50644), and Volume Three (Document No. 50645) contain much the same information as the guides for DOS DATABUS, as it pertains to RMS.

Video Tape Explains DDP

Customer Education has a new video tape called "Concepts of Dispersed Data Processing" to help people who are unfamiliar with the concepts of dispersed data processing understand it a little better.

The tape informs the viewer on the basics of data processing, discussing inputs and output, defining terms, and describing the types of operating systems, application programs, and systems programs used in a typical configuration. The tape deals with questions such as:

* What is a computer?
* How are computers alike, and how do they differ?
* What is data?
* What are printers?
* What are peripherals?
* Why do computers use a variety of programming languages?
* What is the difference between hardware and software?

As you can see from these questions, this tape is directed toward people who do not have a basic concept of what data processing is or how it functions.

The cost of the tape will be $125.00. If you're interested in having a copy, please call Customer Education at 512/341-3268.

Returning Consumables

If you receive damaged consumables, such as ribbons, you can return them to Customer Service for credit or replacement.

To receive credit, call the Customer Support Center at 800-531-5770 (in Texas 800-292-5100; In Hawaii or Alaska, 800-531-5642). Give the coordinator the model code of the item and the reason for the return. The coordinator will issue you a Return Material Authorization (RMA) number. Write the RMA number on the outside of the box of the material being returned to make sure you receive prompt credit or replacement of those items. Send the supplies to the following address:

Datapoint Corporation
981 Isom Road MS#B-52
San Antonio, Texas 78216
ATTN: RMA Coordinator
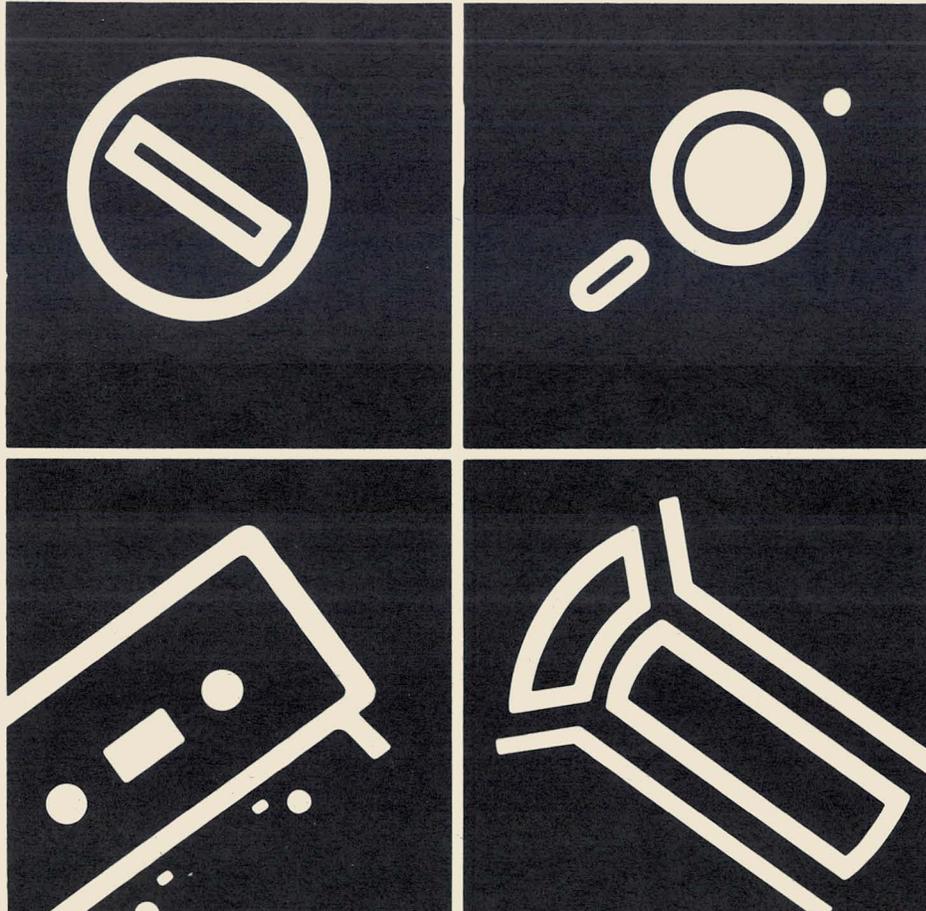
Users Group Formed in Tennessee

The Mid-South Users Group is being formed in the Tennessee area. The group will include users from the surrounding areas as well.

If you're interested in being part of a users group and are close to Tennessee, please write or call:

Mid-South Users Group
c/o Sam Cantor
Arthur Fulmer Computer Services
P.O. Box 177
Memphis, TN 38101
901/525-5711

Seattle Group Has First Meeting

On October 25, the Seattle Users Group held its organizational meeting. The group will hold its meetings in the south central part of Washington. If you're interested in being part of the Seattle Users Group, please call or write:

Seattle Datapoint Users Group
c/o Dianne Lee
Weyerhaeuser Real Estate Co.
Tacoma, WA 98477
206/924-3296

Michigan Datapoint Users Group

If you tried to contact Thomas Kelly for information on the users group in Michigan, I'll bet you had problems. The phone number listed in Issue 20 of *SOURCE DATA* is incorrect. If you're interested in being a part of the Michigan Users Group, please call or write:

Mr. Thomas Kelly
Michigan Datapoint Users Group
7410 Anthony
Dearborn, MI 48216
313/584-7963

TriState Users Group Organizes

During the organizational meeting of the TriState Datapoint User's Group meeting on September 20th, the following people volunteered for positions in the group's structure (unless otherwise noted, companies are located in New York):

President	
Coordinator &Chairman	Ruby Kirk (Scientific American Medicine)
1st Vice President	Harvey Cohen (Combustion Engineer, CT)
2nd Vice President	Joe Zarabet (Lever Bros. Co.)
3rd Vice President	Chuck Bennett (American Cynamid, NJ)
Treasurer	Stu Schwartzreich (Accent Data)
Asst. Treasurer	Mark Goldstein (General Electric, CT)

President — Rodger Kelly (Chase Manhattan)

Committees:	
Program	Bernd Walter (Money Market System)
Public Relations	Bob Putignano (Access Data)
Publications	Jeannie Stark (L.F. Rothschild)
User's Profile	Pat Bunosso (General Electric, CT)
Constitution & By-Laws	Steven Kolman (Lever Bros. Co.)
	Bill Hudson (Edward, Isaccs, & Co.)
	Thomas McNee (Central National Corp.)

"The vice presidents are organized regionally so if membership in one state becomes large enough to form another chapter, a president will be immediately available," says Ruby Kirk, coordinator and founder of the group. Until that time, the vice presidents will serve when the president is not available.

Committees were formed to maintain all areas of the group, including public relations (membership and liason with companies and other groups) and user's profiles (to maintain resumes of group members).

If you would like to become part of the TriState Users Group, please call or write:

TriState Datapoint Users Group
c/o Ruby Kirk
Scientific American Medicine
415 Madison Ave., Floor 20
New York, NY 10017
(212) 754-0473
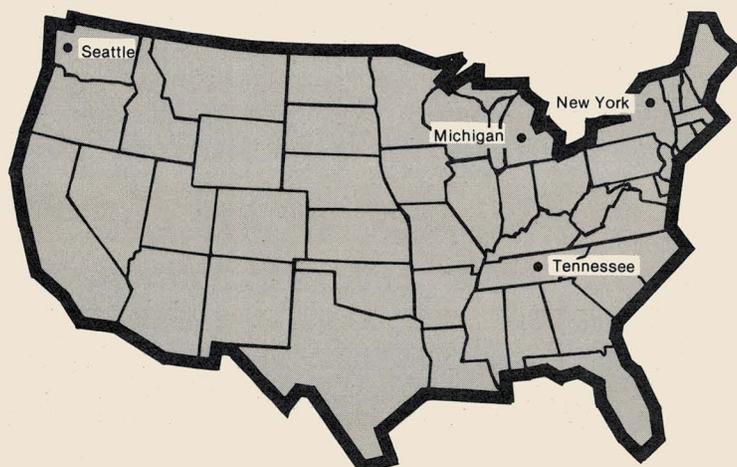
Customer Education Schedule

Below is the schedule for Customer Education classes from November to January. *The Customer Education Course Catalog,* Document No. 60269, contains course descriptions, prerequisites, education center information, and cost. Copies are available at your local Datapoint office. Enrollment for courses is handled by the Customer Education Center; general questions about courses may be directed to the Center by calling (512)341-3268.

Course	San Antonio	San Mateo	Chicago	Washington D.C.	Boston	New York	Seattle
Introduction to Datapoint Programming	10/18 11/15 12/13 1/24	11/15	11/29 1/24	1/17	11/1	11/29 1/31	1/24
DOS Basic Concepts and Operations	11/1 12/6 1/31	11/1 1/17	11/1 1/10			1/24	12/13
DOS Advanced Operations	11/8 12/6 1/17		12/13	12/6		11/1 1/10	1/17
RMS Basic Concepts	11/15 1/3 1/31					12/6	
DOS Basic Word Processing	11/8 12/13 1/17	11/8 1/3 1/31	11/8 12/6 1/3 1/17	11/15 1/10	11/29	11/29 1/17	1/31
DOS Advanced Word Processing	11/15 1/17 1/24						
RMS Basic Word Processing	11/8 11/29						
RMS Advanced Word Processing	11/1 12/13 1/17						
DOS DATABUS	11/1 1/3 1/17	12/6 1/31	11/8 1/3	1/3	12/6 1/24	11/15 12/13 1/10	1/10
RMS DATABUS	1/10						
DOS Advanced DATASHARE	11/29				11/15	1/10	
RMS for New Datapoint Customers	11/1 1/17		11/15	1/24	1/3		12/6
RMS for DOS Customers	11/8 1/10	1/24	11/29	11/1		12/13	
DOS Attached Resource Computer (ARC)	11/15 12/13 1/10		1/24	11/8			
DOS Electronic Message System	1/10						
DOS Automatic Call Distributor (ACD)	12/6 1/17						
DOS Advanced ACD	12/13						
DOS Long Distance Control System (LDCS)	11/29 1/24						
DOS Advanced LDCS	1/3						
DOS Assembler (SNAP)	1/10						

SOURCE DATA Advertising

The advertising listed below is presented without verification or review of any kind by Datapoint Corporation. Datapoint Corporation makes no representation concerning the content of these advertisements or the quality of any product, software, or service offered. Datapoint Corporation reserves the right to reject any advertisement at its sole discretion.

SOURCE DATA Advertising Format

Advertising rates are $22/block. The block below contains the correct number of spaces to fill a one-block ad space (1 x 2 1/4"). Use one space for each character, space, or punctuation mark you want to use. You may supply camera-ready art for any advertisement. All advertisements are restricted to a maximum of 6 blocks per advertisement. Camera-ready art must conform to Datapoint graphic standards. Failure to conform to these standards will result in significant delays of publication and may cause the ad to be rejected at the editor's discretion.

In addition to the text for the advertisement, please complete the rest of the form and return it, with payment, no later than **December 1** for publication in the December/January issue of *SOURCE DATA*.

Classified ad text:

Classified as above: $22 ☐   Display: ☐ No. blocks_____Payment: $22/block $_____

I agree to idemnify Datapoint against liability arising from this ad and that Datapoint's liability for any matter connected with this ad shall not exceed $22.

Your signature _____

Name _____Company _____

Address _____City _____

State_____Zip_____Phone_____

- -

SOURCE DATA Reader Reply Form

This is your space to send us questions, comments, criticisms, ideas, anecdotes, address corrections. Send to the address shown on reverse.

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

DATAPOINT

SOURCE DATA #41
9725 Datapoint Drive
San Antonio, Texas 78284