THE DEC* PROFESSIONAL

• VOLUME 3, NUMBER 1 • THE MAGAZINE FOR DEC USERS

LAST CHANCE T·I·M·E IS·RUNNING·OUT!!!

RENEW·NOW·TO·KEEP·YOUR·FREE·ISSUE COMING·FOR·ANOTHER·FULL·YEAR

PLEASE BE SURE TO SIGN AND DATE BEFORE MAILING POSTAGE PAID CARD

PLEASE DISREGARD THIS NOTICE IF YOU HAVE ALREADY MAILED IN YOUR RENEWAL.

V3.1

THE DEC* PROFESSIONAL
THE MAGAZINE FOR DEC USERS

Please print all information requested. Please answer all questions.

Area Code ____ Office Telephone ____

Do you wish to receive (continue to receive) a **FREE** subscription to The DEC Professional

☐ Yes ☐ No Date X _____

☐ Check here if renewal.

☐ Canada $25 (US$) ☐ Foreign $50 (US$) check must accompany this card.

...re required to qualify does not obligate me or my company.

MAKE ANY NECESSARY ADDRESS CHANGES HERE

NAME _____

TITLE _____ MAIL STOP _____

COMPANY _____

ADDRESS _____

CITY/STATE/ZIP _____

...y area EDP activity (check ONE only)
...uting or Data Processing Equipment
...of Computing or Data Processing Equipment

...OMPANY MANAGEMENT

...**General & Corporate:** President/Owner/Director/Chairman/Partner, Vice President/General Manager.

...**Financial:** Comptroller/Treasurer, Vice President Finance, Chief Accountant.

...**Administrative Services:** Vice President Operations, Director or Manager Purchasing/Personnel/Office, Directors or Managers of Microfilm.

D. **Marketing:** Vice President Sales/Sales Manager, Director or Manager Marketing.

COMPUTER/SYSTEMS OPERATIONS

☐ E. **Management:** Vice President EDP, Director/Manager/Head/Chief of EDP/MIS, Director or Manager of Programming, Director or Manager Systems & Procedures, Director or Manager Communications, Director or Manager EDP Operations.

☐ F. **Supervisory:** Supervisor Equipment/Analysis, Chief/Lead/Senior Programmer, Chief/Lead/Senior Mathematician, Senior Systems Analyst.

☐ G. **Staff:** Analyst/Systems Analyst, Mathematician, Programmer, Machine or Computer Operator, Software Designer, EDP Auditor.

ENGINEERING/MANUFACTURING

☐ H. **Management:** Vice President or Director of R&D, Vice President or Director Engineering, Plant or Production Manager/Director, Chief Engineer.

☐ I. **Staff:** Member of Technical Staff/Group Leader, Systems Engineer, Design/Production/Research Engineer, Maintenance Engineer, Application Engineer, Other Engineer.

☐ J. **CONSULTANT**

☐ K. **EDUCATOR:** Educator (instruct EDP) and Educator (use EDP)

☐ L. **OTHER** _____

...Primary end product or service of your firm at ...s location. (Insert code)

A. Manufacturing: Computers, Data Processing and Peripheral Equipment.
B. Other Manufacturing and Processing.
☐ C. Government: Federal, State, Municipal, including Military.
☐ D. Educational: Colleges, Universities, and other Educational Institutions.
☐ E. Computer and Data Processing Services including Software Services, Service Bureaus, Time Sharing and EDP Consultants.
☐ F. Finance: Banking, Insurance, Real Estate, Securities, and Credit.
☐ G. Trade: Wholesale and Retail.
☐ H. Medical and Legal Services.
☐ I. Transportation Services: Land, Sea, and Air.
☐ J. Utilities: Communications, Electric, Gas, and Sanitary Services.
☐ K. Construction, Mining and Agriculture.
☐ L. Other Business Services, such as Business Management Consultants, Hotels, Amusements and Non-Profit Organizations.

Total number of employees in the entire company/institution/governmental department.
☐ 1. 10,000 or more ☐ 2. 5,000 to 9,999 ☐ 3. 2,500 to 4,999 ☐ 4. 1,000 to 2,499 ☐ 5. 500 to 999 ☐ 6. 250 to 499 ☐ 7. 100 to 249 ☐ 8. 6 to 99 ☐ 9. Fewer than 6 (Check one only)

CPU (enter any you have)
...AX 782 ☐ 1
780 ☐ 7
750 ☐ 14
730 ☐ 13
...ROVAX ☐ 18
...DP 70 ☐ 31
44 ☐ 23
34 ☐ 24
PDP 24 ☐ 25
23+ ☐ 21
23 ☐ 28
35/40/45/60, etc ☐ 30
MICRO-11 ☐ 16
LSI-11 03 ☐ 42
11/73 ☐ 44
LSI-11 Falcon ☐ 46
T-11 ☐ 48
PDP8 ☐ 51
WPS ☐ 61
VT180 ☐ 63
10/20 ☐ 64
PDP(other) ☐ 65
PROFESSIONAL 350 ☐ 73, 325 ☐ 74
RAINBOW 100 ☐ 83, 100+ ☐ 88
DECMATE I ☐ 81, II ☐ 86

6. DISK TYPES (check any)
RX ☐ 101
RL ☐ 102
RC ☐ 106
RK ☐ 107
RP ☐ 108
RM ☐ 111
RA ☐ 113
RF/RS ☐ 119
other ☐ 118

7. DISC SUPPLIER (check either or both)
DEC ☐ 125
NON/DEC ☐ 131
I also have
HSC50 ☐ 132
DECNET ☐ 133
ETHERNET ☐ 134
STREAMERS ☐ 135
TU/TE TAPES ☐ 136
DBMS ☐ 137

8. TOTAL DISK STORAGE (check one)
<5MB ☐ 1
5-10MB ☐ 2
11-100MB ☐ 3
101-200MB ☐ 4
201-500MB ☐ 5
501-1GB ☐ 6
>1GB ☐ 7
>2GB ☐ 8

9. OP/SYS (check any you use)
VAX/VMS ☐ 1
RSTS/E/CTS500 ☐ 2
RSX-11 ☐ 3
RT11/CTS300 ☐ 4
MUMPS ☐ 5
P/OS ☐ 6
CP/M ☐ 7
TSX ☐ 8
UNIX ☐ 9
TOPS ☐ 10
IAS ☐ 11
MS/DOS ☐ 12
CONCUR CP/M ☐ 13

10. I am an OEM ☐ YES ☐ NO

#43042 DEC V3.1
BILL PITTS
MINI COMPUTER APPLICATION
730 MORA DR
LOS ALTOS CA 94022

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 16 AMBLER, PA

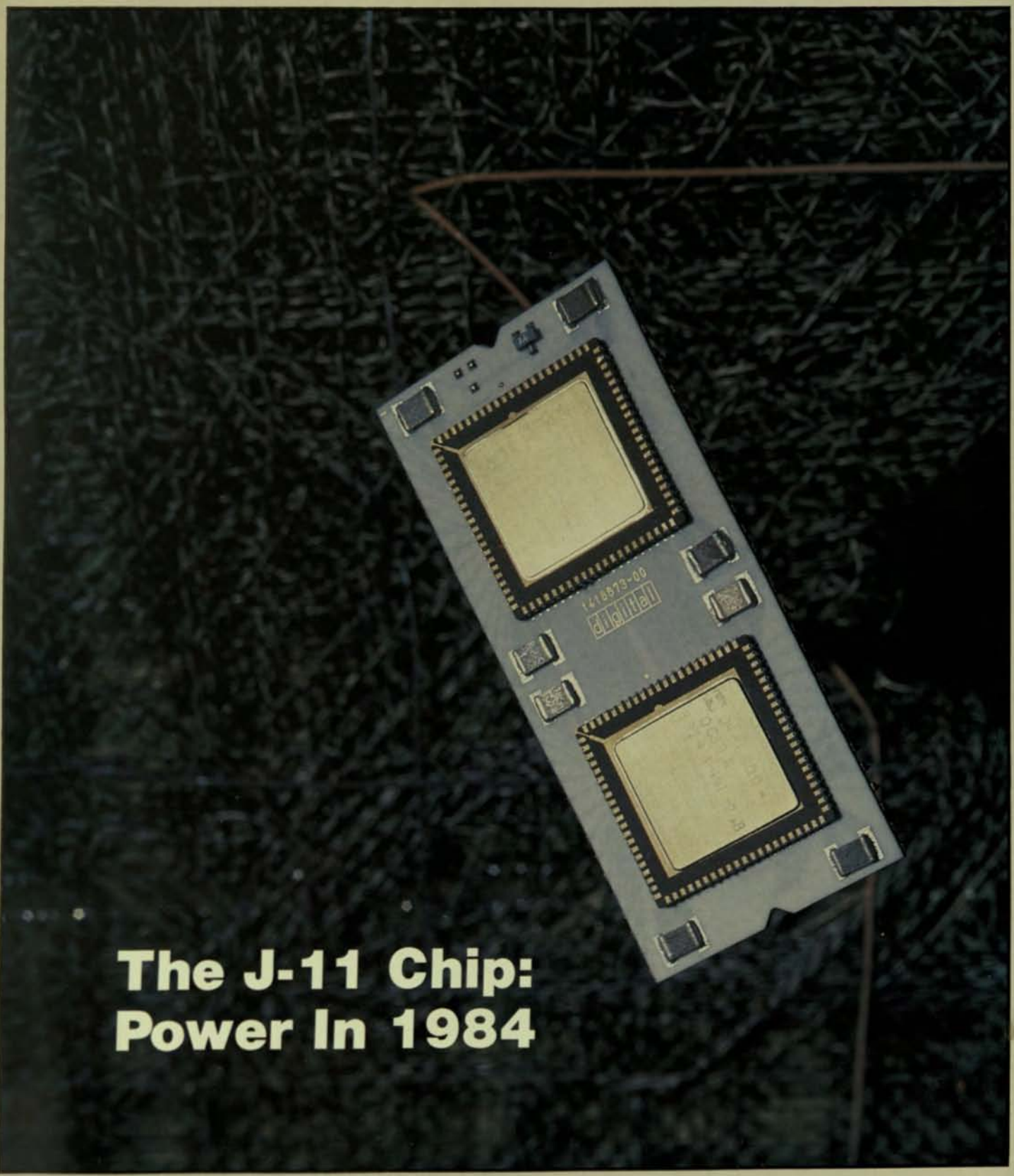
POSTAGE WILL BE PAID BY ADDRESSEE

THE DEC PROFESSIONAL
P.O. BOX 362
AMBLER, PA 19002-0362

JANUARY • 1984

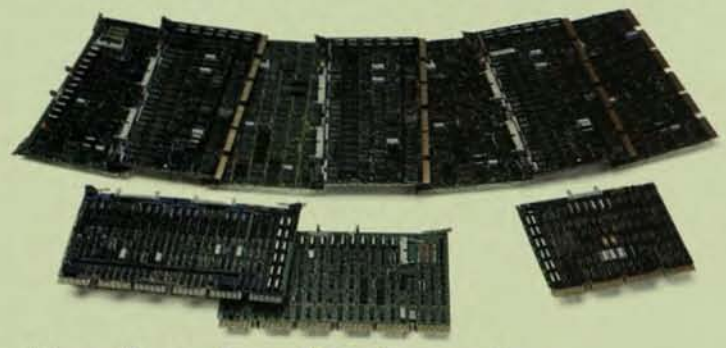
THE DEC* PROFESSIONAL

• VOLUME 3, NUMBER 1 • THE MAGAZINE FOR DEC USERS



The J-11 Chip: Power In 1984

Five reasons why DEC users should buy Emulex communications controllers.



Broad product line featuring our new DMF-32 emulation.

Nobody covers LSI-11, PDP-11, and VAX-11 users' needs like Emulex. More than 15 software-transparent controllers emulating DH11, DZ11, DV11 and DMF-32. All deliver improved line-handling capabilities, in a smaller package, at lower costs.

More channels.

Emulex's new DMF-32 emulation is typical. One controller board handles up to 64 lines, vs. only eight per DEC module. And Emulex offers *all* lines with modem control, not just two. For even more lines, Emulex's Statcon Series is the answer. We simply add a low-cost port concentrator, so that with one controller board you can connect up to 256 remote *and* local terminals.

Easy growth path.

As your system grows, upgrading is simple with Emulex controllers. Just change PROM sets. Example: DH to DMF for $350. In addition, Emulex's advanced microprocessor architecture is consistent throughout the product line. Think of the inventory savings.

Fewer backplane slots.

Emulex communications controllers pack so much capability onto each board that fewer boards are needed. Take a 64-line DH11 emulation. Emulex does on one board what it takes DEC to do on 36. Think of the savings in rack space, to say nothing of price.

Lower prices.

For instance, a DEC DH11 controller lists at $8,950 per 16 lines, with expansion chassis costing $3,000 or more. Compare that to Emulex's CS11/H at $4,500 for the first 16 lines and $3,000 for each additional 16 lines. At 64 lines, you suddenly have savings of about $23,000 and a lot of extra slots to boot.

Don't speculate with your communications controller dollars. Invest in Emulex. Phone toll free: (800) 854-7112. In California: (714) 662-5600. Or write: Emulex Corporation, 3545 Harbor Blvd., P.O. Box 6725, Costa Mesa, CA 92626.

EMULEX

The genuine alternative.

GSA Contract #: GSOOK8401S5575

CIRCLE D107 ON READER CARD

Our IBM Protocol Converter is not the same under the skin.

MICOM's new Micro7400 is very, very different. It not only provides a Gateway for dumb asynchronous terminals to access IBM mainframe applications, but goes far beyond the basic capabilities of IBM 3270s and other protocol converters. *It's even easy to use.*

Since MICOM is the world's largest volume manufacturer of data concentrators—thanks to its Micro800/2 "Orange Box" family—it's only natural that MICOM's protocol converter should include the same features and capabilities that made the data concentrators so popular. And fortunately for asynchronous terminal users wishing they could connect to IBM mainframes, and for mainframe DP managers looking for less expensive terminals, adding concentrator features to a converter makes for a surprisingly effective hybrid.

All Standard IBM 3270 Features

Functioning as an IBM 3274 Model 51C Cluster Controller using either Bisync or SNA/SDLC protocol, the Micro7400 allows ASCII terminals (or personal computers emulating terminals) to perform as 3270s. Display terminals emulate IBM 3278s; printers emulate IBM 3287s. And special software allows printer terminals to interact with full-screen programs originally developed for crt's.

Plus Extra Functions

The Micro7400 also offers features not available in the IBM 3270 line, including dial-up access to the protocol converter, terminal-controlled diagnostics, and do-it-yourself channel configuration for setting terminal-related parameters like parity.

Support for IBM Personal Computers

IBM PCs connected to the Micro7400 can emulate IBM 3270 terminals, too, for communicating with mainframes. Diskette-based MICOM software makes it easy.

Switching Between Multiple Hosts

Users can switch between two IBM hosts, or between an IBM host and asynchronous ports on one or more minicomputers—completely under terminal control.

Command Port

Unique among protocol converters, the Micro7400's Command Port allows a network manager to dynamically alter operating parameters like priority assignment, as well as providing monitoring, diagnostic, and control facilities.

As Low As $400 Per Channel

Even the pricing is more like a concentrator. Standard models are available to support from two to 12 channels, at prices as low as $400 per channel. A budget-minded 2-channel "LTD" model is also offered, as are cost-saving versions with built-in modems. And there's much more. Call today for a price list and a 12-page color brochure, or use the Reader Service Number below.

For protocol conversion without tears, think MICOM.

MICOM

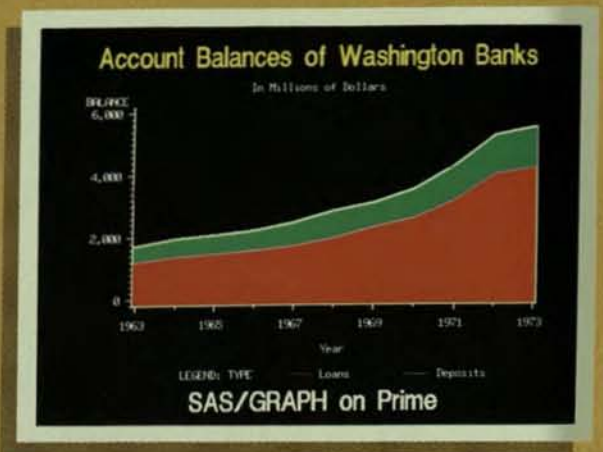
MICOM SYSTEMS, Inc. • 20151 Nordhoff Street • Chatsworth, CA 91311 • Telephone (213) 998-8844 • TWX 910/494-4910
Regional Sales/Service • Atlanta, GA • (404) 435-2999 • Boston, MA • (617) 527-4010 • Chicago, IL • (312) 789-2430
Dallas, TX • (214) 258-0774 • St. Louis, MO • (314) 576-7626 • Teaneck, NJ • (201) 836-4000
MICOM-BORER Ltd. • Bel Court • 15 Cradock Road • Reading, Berkshire RG20JT, England • (0734) 866801 • Telex 847135

Available *now* from these stocking reps...

AK: Anchorage (907) 561-1776 / Juneau (907) 789-4101 ● **AL:** (800) 327-6600 ● **AR:** (214) 620-1551 ● **AZ:** (602) 994-5400 ● **CA:** Anaheim (714) 635-7600 / Lodi (209) 334-1961 San Diego (619) 565-1557 / San Jose (408) 298-7290 ● **CO:** Colorado Springs (303) 594-0880 / Denver (303) 777-8070 ● **CT:** (617) 235-5520 ● **DE:** (609) 779-0200 **FL:** (800) 432-4480 ● **GA:** (800) 327-6600 ● **HI:** (808) 537-9758 ● **IA:** (402) 895-5850 ● **ID:** (801) 466-6522 ● **IL:** (312) 255-4820 ● **IN:** (317) 846-2591 ● **KS:** (816) 252-3700 **KY:** (502) 228-5401 ● **LA:** (800) 327-6600 ● **MA:** (617) 235-5520 ● **MD:** (301) 261-4344 ● **ME:** (617) 235-5520 ● **MI:** (313) 588-2300 ● **MN:** (612) 425-4455 ● **MO:** Independence (816) 252-3700 / St. Louis (314) 721-0401 ● **MS:** (800) 327-6600 ● **MT:** (801) 466-6522 ● **NC:** (800) 327-6600 ● **ND:** (612) 425-4455 ● **NE:** (402) 895-5850 ● **NH:** (617) 235-5520 **NJ:** North (201) 569-2353 / South (609) 779-0200 ● **NM:** Albuquerque (505) 292-1212 / Las Cruces (505) 524-9693 ● **NV:** (714) 635-7600 ● **NY:** Albany (518) 459-5891 Buffalo (716) 662-4568 / New York City (201) 569-2353 / Rochester (716) 442-5631 / Syracuse (315) 638-2042 ● **OH:** Cleveland (216) 524-5930 / Dayton (513) 434-7500 **OK:** (405) 478-5000 ● **OR:** (503) 224-3145 ● **PA:** East (609) 779-0200 / West (412) 892-2953 ● **RI:** (617) 235-5520 ● **SC:** (800) 327-6600 ● **SD:** (612) 425-4455 **TN:** (800) 327-6600 ● **TX:** Dallas (214) 620-1551 / Austin (512) 327-8600 / El Paso (915) 542-1762 / Houston (713) 353-7728 ● **UT:** (801) 466-6522 ● **VA:** (301) 261-4344 **VT:** (617) 235-5520 ● **WA:** (206) 454-2383 ● **WI:** (414) 784-9379 ● **WV:** East (301) 261-4344 / West (412) 892-2953 ● **WY:** (303) 777-8070 ● **Washington, DC:** (301) 261-4344 **Puerto Rico:** (809) 723-9689

CIRCLE D111 ON READER CARD

SAS, Not Just for IBM ...Anymore



Since its beginnings, SAS® has provided a powerful and flexible software system to IBM users. Now SAS offers VAX,™ ECLIPSE® and PR1ME® users the same power and flexibility.

You get a twofold advantage with the SAS System. A library of ready-to-use procedures gives you the power to analyze data without formal training. SAS even provides an on-line help facility. With a few English-like commands, you can use SAS for data analysis, market research, capacity planning, financial reports, summary statistics, charts, plots, personnel reports and many other jobs.

In addition to these routine reports, the SAS System provides extensive data management and retrieval tools, combined with a macro facility, to handle all your complicated programming needs.

In short the SAS System gives you the analytical tools for a complete Decision Support System, integrating SAS for data management, statistical analysis and report writing; SAS/GRAPH™ for color graphics; SAS/ETS™ for econometric and financial modeling, and forecasting; and SAS/FSP™ for data entry, letter writing and financial spreadsheet reporting.

Now you can enjoy the power and flexibility of mainframe software on a minicomputer—with SAS.

Call or write today. SAS Institute Inc., SAS Circle, Box 8000, Cary, NC 27511 USA. Telephone (919) 467-8000. Telex 802505.

SAS Institute GmbH,
Rohrbacher Strasse 22, D-6900
Heidelberg 1, West Germany.
Telephone 06221-29014. Telex 461581.

SAS Software Limited,
The Centre, 68 High Street,
Weybridge, Surrey KT13 8BL,
UK. Telephone 0932-55855.
Telex 946138.

SAS Institute (NZ) Ltd.,
Level 9, Greenock House, 39 The Terrace, PO Box 10-109,
Wellington, New Zealand. Telephone (04) 727-595. Telex NZ 31525.

SAS Software Pty. Ltd.,
Level 60, MLC Center, 19-29 Martin Place, GPO Box 4345, Sydney,
NSW 2001, Australia. Telephone (02) 235-2199. Telex AA 71499.

CONTENTS

DEPARTMENTS

FROM THE PUBLISHERS . . .

WINDOWS ON THE WORLD

Dave Mallery

COMDEX Las Vegas — seems I was just here for DEXPO. There are a few differences however . . . You could lose five DEXPO's there and still have room for the sum of all the computer shows ever given. They were claiming an attendance of 105,000. Hundreds of miles of aisles. All of them jammed. You had to see it to believe it.

Was it too big? Not really. The second industrial revolution is worthy of such a send-off.

Remarkable new hardware? Not really. (There were a few, but just a few.) Mature hardware? No, just adolescent. But in between the zits and the peach-fuzz, you can begin to see the adult emerge.

Amazing software? Some, but mostly tons of re-worked mini accounting systems that don't do justice to their new-found medium. In some cases, however, the software begins to out-strip the hardware (that's a switch!). Both have a long way to go.

It's clear to me that multi-megabyte, megaflop machines will be needed just to support the next generation of software. Which brings me to windows.

I used windowing for the first time in 1975. Hard at work on development of my first major RSTS software project, I installed two terminals on my desk. Even though I had an unloaded '70 on line, I was able to keep them both quite busy (not to mention the printer) in my office . . . Time, economics (terminals were quite expensive in '75) and other duties caused me to relinquish my other window, but it was great.

I didn't even need a mouse!

It seems that the buzz word for the beginning of 1984 is 'windows'; that is, a scheme to break your display into small segments, each trying to display one of the concurrent tasks you are running. There is one fundamental problem. Windows don't really work on a screen that is 80 by 24. There is not enough resolution . . . put another way, by the time you get the third window going, there is not enough information in any one window to really do more than a snazzy demo.

Look at the VAXstation. The difference is the size of the monitor and its resolution. Windowing works on such a display because the overall bandwidth of the screen is several times that of your friendly VT100.

The problem is that bandwidth is still inexorably expensive. This is an era in which the resolution of the DEC PC monitor is the best in the industry, yet is only as good as your garden variety terminal. The DEC monitor and keyboard have "pushed" the PC industry and IBM hard simply because of their bandwidth (keyboard ergonomics is a form of bandwidth!).

It's a little like hi-fi (remember that?). The best components in the world are limited by the bandwidth of the transducer that ultimately puts the sound in the room (loud speaker). Someone is going to have to spend quite a few megabucks on the problem. The rewards will follow.

The next winner will be the company that can mass produce a monitor with the resolution of the VAXstation and the price of a VT220. This is probably quite attainable in a world that can produce 200MB 5 1/4 inch Winchesters that sell for under three thousand dollars, by the thousands. (Yes, Virginia, they were at COMDEX too!)

However, it's just about 1984 and time waits for no one.

IT'S TIME TO STOP PROMISING AND START DELIVERING

Carl B. Marbach

1984 will be the year of the VAX . . . The J-11 chip will give us 11/70 power on an 11/23 sized computer . . . The Professional 350 computer is as professional as you are . . . VAX clusters will be fully software supported . . . RA60 disk drives will be shipped in volume . . . The UDA-50 controller problems have been worked out . . . The RC25 disk drive . . . The JUPITER project . . . BASIC PLUS-2 on the Professional 350 . . . the Micro-11 . . . the MicroVAX.

Promises, promises.

Have you heard these (or any other) promises from your favorite computer company lately? Most of us have been hearing things like this for some time now. "I can't wait for the future," a corporate executive told me recently, "we need the computer power now." DEC's future always seems to be just beyond the horizon.

While we know that the VAX 11/780 is an outdated machine, we are doomed to buy it until either (1) a new VAX comes out that replaces the aging 780, or (2) wait until VAX clusters are fully supported by VMS. Option two is probably the closest to reality with V4.0 scheduled for about the middle of the year. Well, it's scheduled for March but the middle of the year is a better guess. How can loyal customers be sold computers that are second rate in today's market?

Promises, promises.

. . . continued on page 57

THE DEC PROFESSIONAL
THE MAGAZINE FOR DEC USERS

Publishers: R.D. Mallery, Carl B. Marbach

Managing Editor: James L. Trichon

Director of Advertising: Helen B. Marbach

Business Manager: Peg Leiby

Advertising Sales Manager: Hope Makransky

Assistant Editor: Linda DiBiasio

Production Manager: Georgia F. Conrad

Subscription Fulfillment: Kathi B. Campione, Claire Hollister, Margie F. Pitrone, Connie Mahon, Terry McMahon, Donna L. Schmidt

Contributing Editors: Arthur Edward Groulx, Steve Holden, Richard Lowe

Contributors: Stephen L. Arnold, Tom Bandy, Frank R. Borger, Jeffrey H. Gove, John Gram, Rex Jaeschke, Norm Kiefer, Austin Kinsella, Charles Mansfield, Arthur Metz, Bob Rennert, Terry C. Shannon, Anthony Stracciolini, Robert J. Walsh, Allen A. Watson, James B. Wilkinson

Cartoons: Frank Baginski, Douglas Benoit, Emily Dahlstrom, Johns, John K. Young

Colorado Representative:
Jeff Yusem
P.O. Box 8013, Aspen, CO 81612
303/920-2180

Mid-Atlantic Representative:
Ed Shaud
P.O. Box 187, Radnor, Pa. 19087
215/688-7233

United Kingdom Representative:
Pauline Noakes
RTZ Computer Services, Ltd.,
P.O. Box 19, 1 Redcliff Street
Bristol, BS99-7JS
Phone: Bristol 24181

Design & Production: Grossman Graphics

Printing & Binding: Mack Printing

Cover Art: Jim Lang

Circulation: 75,000 DEC users, worldwide

Editorial Information: We will consider for publication, all submitted manuscripts and photographs, and welcome your articles, photographs and suggestions. All material will be treated with care, although we cannot be responsible for loss or damage. (Any payment for use of material will be made only upon publication.)

THE DEC PROFESSIONAL Magazine, January 1, 1984, Vol. 3, No. 1. ISSN 0744-9216 is published bi-monthly by Professional Press, Inc., 161 E. Hunting Park Avenue, Philadelphia, PA 19124. Subscriptions: no charge for qualified U.S. sites, foreign - $50 (US$) per year. Second Class postage paid at Philadelphia, PA., and additional mailing offices. POSTMASTER: Send all correspondence and address changes to: THE DEC PROFESSIONAL, P.O. Box 362, Ambler, PA 19002-0362, telephone (215) 542-7008. BPA membership applied for November, 1982.

OUR MICRO/11 STANDS ALONE

You know about the advantages of a Micro/11 computer system. What you may not know is that it's available now. Our MDB Micro/11 is functionally equivalent to the DEC Micro/PDP-11* providing an 11/23 Plus, 256KB RAM, 10.4 MB Winchester and 1 MB Dual Floppy sub-system. But there's more.

This low-cost, compact and highly flexible work station provides the exclusive feature of being software driver and media compatible to the RX02. This unique capability allows diskette transfer to and from other DEC systems. Also, unlike the DEC unit, our Winchester is RL02 software compatible. Even optional 20 MB RL02 or RP02 emulating Winchesters are available to enhance your system.

When it comes to interface modules, however, the MDB Micro/11 has lots of company. The system, with its 8 quad slot (16 dual slot), Q-22 backplane and its rear distribution panel, accommodates all of MDB's unequalled repertoire of FCC compliant Q-bus controllers and interfaces. They include multiplexors, line printer controllers, disk and tape controllers, high speed DMA modules and interprocessor-links.

FOR ITS AVAILABILITY AND CAPABILITY

As for price, we won't hold you up there either. Single units cost only $7,800 and substantial discounts are available for quantity purchases.

So why wait? It's all available now. Start by contacting us today. You won't be alone.

*Trademark of Digital Equipment Corporation.

MDB SYSTEMS INC. *THE WORLD'S LARGEST INDEPENDENT MANUFACTURER OF COMPUTER INTERFACES.*

Available on GSA contract #GSOOK8401S5502

Corporate Headquarters
1995 N. Batavia Street, Box 5508
Orange, California 92667-0508
714-998-6900 TWX: 910-593-1339 FAX: 714-637-4060

MDB Systems Europe, Inc.
9 route des Jeunes
CH-1227 Geneva (Switzerland)
Tel. (41) (22) 439410 Telex 421341 mdb ch
FAX (41) (22) 439414

MDB Systems, U.K., Ltd.
Everitts House
426 Bath Road
Slough, Berkshire (England) SL1 6BB
Tel. (06286) (67377) Telex (847185) WWTSLO
FAX (41) (2812) (3507)

CIRCLE D232 FOR Q-BUS. CIRCLE D233 FOR MICRO/11.

LETTERS TO THE DEC PRO . . .

Send letters, comments, photos, etc. to: Letters to the DEC PRO, Box 362, Ambler, PA 19002.

In the September 1983 issue of *THE DEC* PROFESSIONAL*, you carried an article on Digital's new personal computer by Carl Marbach. In it he made several statements about the DEC Professional 300 series. One was that each of the personal computers comes with VT102 emulation built in. We have been unable to figure out how to use that capability. According to our documentation you must also have the communications software package before you can use that ability.

He also mentioned that there were several suppliers of facilities to format virgin disks on these computers. I would greatly appreciate receiving the names of these suppliers.

Thank-you very much for the informative, interesting, magazine.

Lloyd E. Sampsel, Director
Educational Computing
Walla Walla College
College Place, WA

DEC promised that all the PCs would have VT-100 emulation. Unfortunately only the Rainbow has it free, the others cost money.

Try MDG & Associates, Moore Park, CA or Silicon Valley Corp., Sunnyvale, CA for RX50 disk formatters.

* * *

I would like to correct a few technical points from "The Image Intensifier for Public Relations," by J. Peter Bryce in the November '83 *DEC PROFESSIONAL*. Television is only 525 scan lines, not 1024. The word for a point on the screen is pixel, not pictel. It is image synthesis and solids modeling software which produces commercial imagery on television, not CAD/CAM, an application process of drafting and milling. The latest CRTs have a resolution of 4096 x 4096, not 4096 x 1024. Image arrays are in 2-D; the data base the image is derived from is a 3-D list. The method of "in-betweening" referenced by Mr. Bryce is more difficult than generally imagined due to occlusion in 3-D space when the object (like the face referenced) is turned part way from the viewer.

An excellent introduction of computer graphics techniques may be found in "Fundamentals of Interactive Computer Graphics," by J. D. Foley and A. Van Dam, published by Addison-Wesley in 1982. The most accepted computer graphics industry society is ACM/SIGGRAPH, which publishes an excellent journal. They may be contacted at 11 West 42nd Street, New York, New York 10036.

Greg Passmore
Principal Software Engineer
Aydin Controls
Fort Washington, PA

* * *

Some time ago a colleague passed along a copy of your publication for my information. I have found it most interesting and useful.

Perhaps some of your advertisers have a DEC mini system with a MUMPS operating system for sale. I am interested in purchasing such a system for my personal use.

James R. Prine, D.V.M.
Los Alamos, NM 87544

* * *

After reviewing a copy of your magazine (March 1983, Vol. 2, No. 2) I wish to complement the staff of *THE DEC* PROFESSIONAL* for putting together such a fine publication on DEC equipment and software; there are a few others doing what you are, but none with the quality and expertise shown in your periodical.

It is because of the quality of your publication that I wish to obtain a subscription for use as a reference in our business. At the present time I am employed by Professional Data Services, a timeshare-service bureau that uses a DEC PDP-11/44 with RSTS/E version 7.2. Many of the articles that have appeared in your publication would be of great interest to us, and would help inform us on new ideas and trends in the industry.

If it would be possible to obtain a complementary subscription to "The DEC Professional," please consider this a request for one; if it is not possible to receive a complementary subscription, please send us the necessary material to purchase a subscription. Thank you.

Kevin S. Carlson
Customer Service Rep.
Havre, MT

THE DEC PROFESSIONAL is distributed free of charge to qualified DEC sites within the US and by paid subscription in Canada and other countries. Your free subscription application is on its way to you. Any readers not currently on our subscription list may call or write for a qualification card. Incidently, your request does not obligate you or your company.*

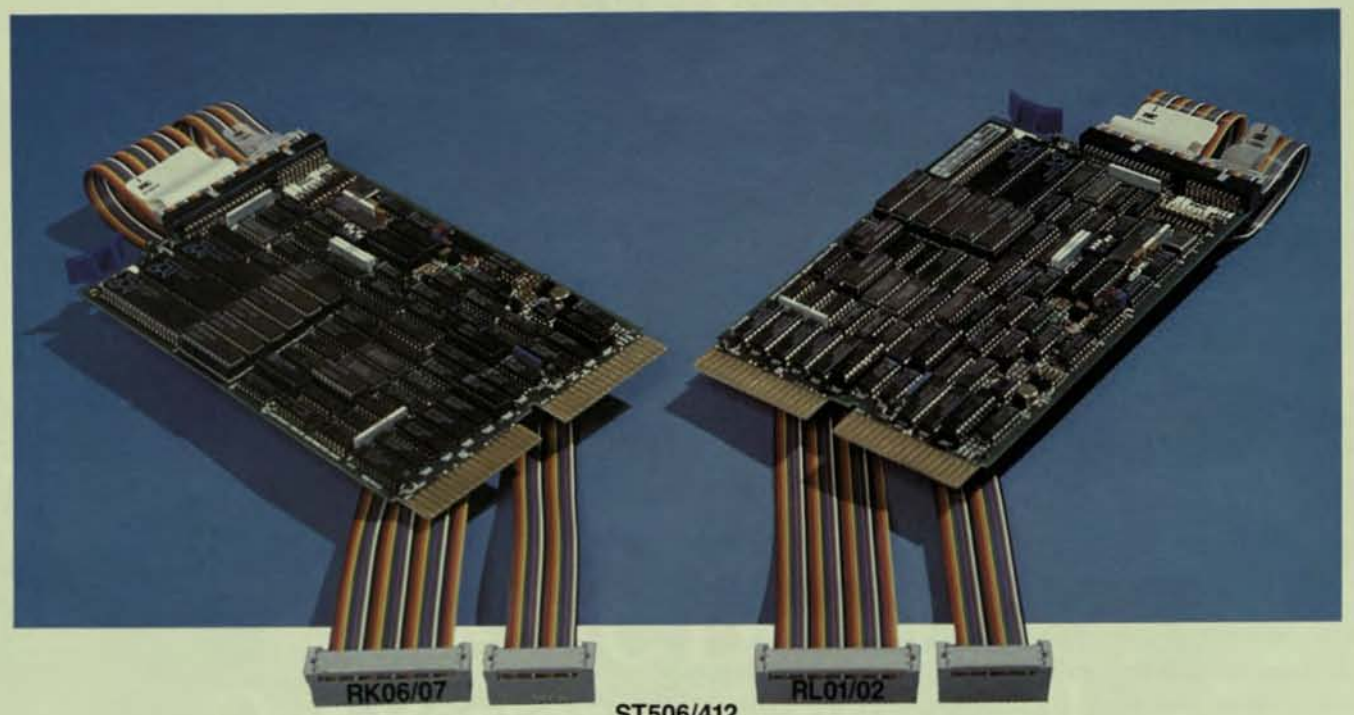
* * *

I am a subscriber to *THE DEC* PROFESSIONAL* and I find it quite informative; I do regret it's not a monthly publication though.

We operate under RT-11 and CTS-300 and all of our applications are written in DIBOL. I have been programming in DIBOL for several years now. I have found the language to be quite easy to use; however, as you know it has limitations. My recent interest has been in developing MACRO subroutines which may be called from a DIBOL program running in TSD. While I am not an experienced MACRO programmer, I am willing to read, research and attempt almost anything. My immediate problem is that none of the reference manuals which I have seen tell you how to do this. There is a lot of information on calling the SYSLIB subroutines from

. . . continued on page 122

The Only Way You Can Get Q-BUS* Compatibility And Interface Flexibility With ANY or ALL These 5¼" Winchester Drives is With These Two Controllers



ST506/412

AMPEX	*FUTITSU*	*MINISCRIBE*	*SEAGATE*
ATASI	*IMI*	*MPI (Micro Peripherals)*	*SHUGART*
ATHENAEUM	*MAXTOR*	*NEC*	*SYQUEST*
CDC	*MEMOREX*	*NISSEI SANGYO*	*TANDON*
CMI	*MICROPOLIS*	*PRIAM*	*TOKYO ELECTRIC*
COGITO	*MICROSCIENCE*	*QUANTUM*	*TULIN*
DISCTRON (RMS)	*INTERNATIONAL*	*RODIME*	*VERTEX*

THE ONLY WAY!

You can't get it with DEC. Or any other independent controller manufacturer.

That's because these controllers feature DILOG's exclusive Universal Formatting.™ That means you can mix or match any two compatible drives (ST506/412), regardless of heads, parameters, capacity, etc. This powerful feature offers you real flexibility in drive selection now...and expansion in the future. It's simple as unplugging one drive and plugging in another.

Model DQ615
- RK06/07 compatible
- Formatted drive capabilities 222.4 MB
- Maximum 8 logical units—two physical

Model DQ614
- RL01/02 compatible
- Formatted drive capacities 41.6 MB
- Maximum 4 logical units—two physical

They both have enhanced 32-bit ECC, 22-bit addressing and RT-11, RSX-11, RSTS and TSX-Plus.

DILOG Universal Formatting™ is also offered for the largest SMD I/O drives. And for your back up there are controllers for ½" tape, ¼" cassette and 5¼" and 8" floppy drives.

Contact your nearest sales office, distributor or DILOG directly for complete data on the way to unmatched Q-BUS compatibility and drive flexibility...THE ONLY WAY!

DILOG

DISTRIBUTED LOGIC CORPORATION

12800 Garden Grove Blvd. ■ Garden Grove, CA 92643 ■ (714) 534-8950 ■ TLX 681399
64-A White Street ■ Red Bank, NJ 07701 ■ (201) 530-0044
12 Temple Square ■ Aylesbury, Buckinghamshire, England HP 20 2QL ■ 44 (0296) 84101 ■ TLX 837038

CIRCLE D115 ON READER CARD

*Trademark of Digital Equipment Corp.

FORTRAN
THE FORGOTTEN LANGUAGE

By Jeffrey H. Gove, James W. Sewall Company, Old Town, Maine

If you have ever felt you were a dying breed, completely forgotten by the RSTS community simply because you program in FORTRAN — take heart — someone cares! It is to you FORTRAN programmers who do not want to give up the speed and familiarity of FORTRAN but would like to do some of the nice things that are available with other languages on RSTS/E, that this article is directed.

FORTRAN IV offers few of the "built-in" enhancements that other languages such as BASIC-PLUS do. Routines to do system calls, string manipulations and simple single character ("ODT-mode") I/O to the user's terminal have not been included in the FORTRAN library routines. However, DEC has provided a fast and efficient means to arrive at these ends via MACRO-11 assembly language.

It is not my intent to teach MACRO-11 here; there are too many other good sources on the subject (Eckhouse, 1979; Gill, 1978; Singer, 1980) and quite frankly, I am by no means an expert. What I do intend to do is stir up enough interest so that you might try one or two of the routines described here, and in addition provide the rudiments which will allow you to code and call assembly language routines from FORTRAN as easily as calling another FORTRAN routine. Incidentally, much insight can be gained into how subroutine linkages are established to FORTRAN subprograms by understanding how FORTRAN to MACRO linkages are accomplished.

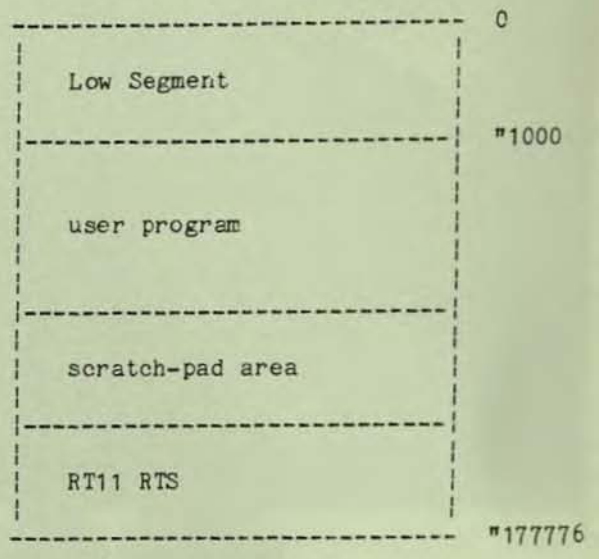


FIGURE 1. The (Simplified) User Job Area

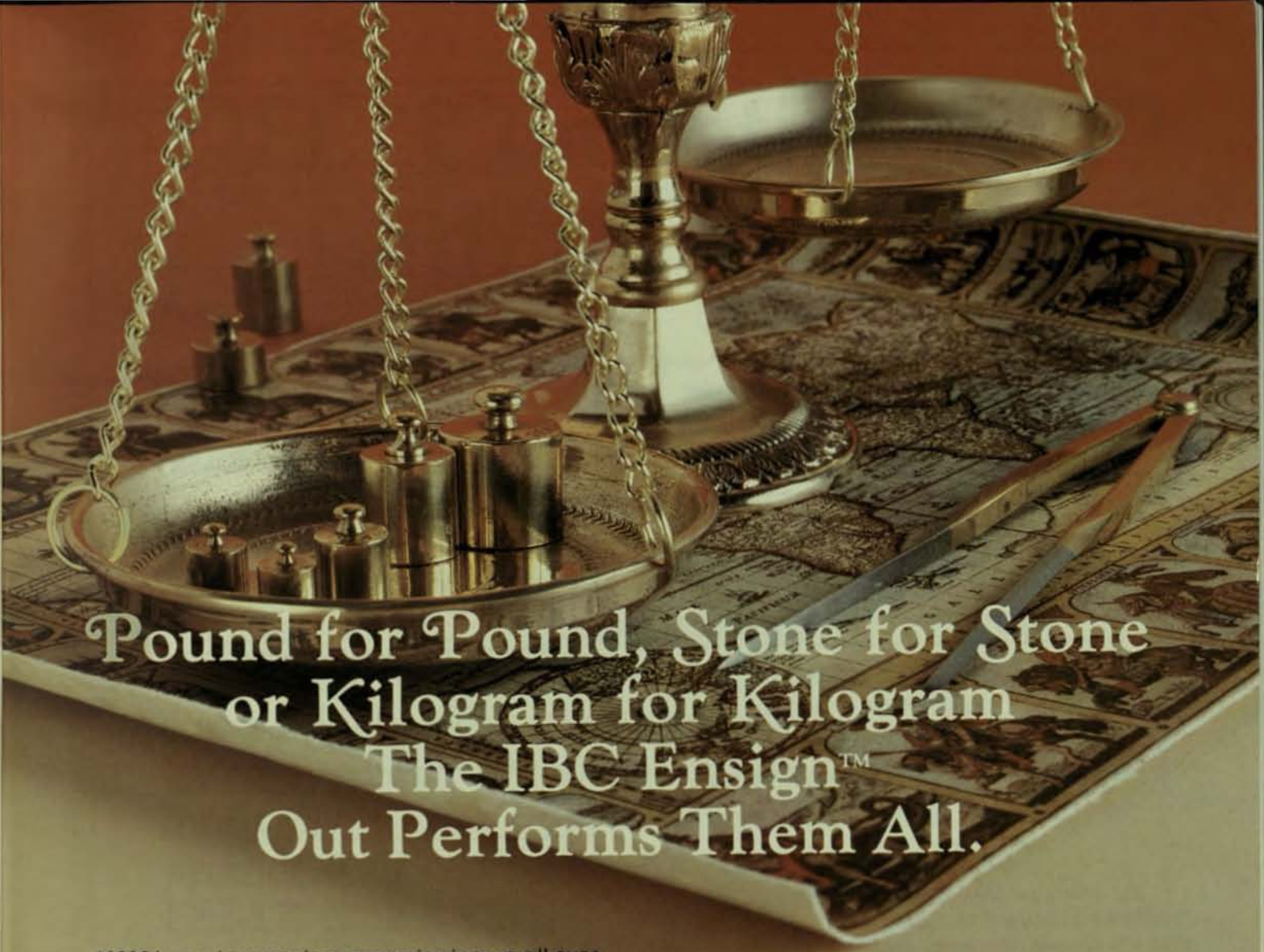
The RT11 RTS and the JSW

Regardless of which run-time system is permanently resident (RSX, BASIC-PLUS, DCL or RT11) as your keyboard monitor, the RT11 run-time system is always loaded into the high segment of your user job area when you run a FORTRAN program. The user job area is a part of memory which is 32K words in size. Of this 32K words, 4K words in the high segment are taken up by RT11 and 1K is taken up by the "scratch pad" area (one place where the user program and RT11 can exchange information). The low segment of the user job image, the first "1000 bytes of virtual address space ("1000 is read octal 1000), is also used by RT11. The rest of the area, about 27K words, is available for the user program as shown in Figure 1.

One word of particular interest in the "1000 byte low segment of the user job image is the Job Status Word (JSW). The JSW is located in bytes "44 and "45 of the low segment. Many of the bits in this word are used by the RT11 run-time system as flag bits to determine what options are currently in effect. Some of these bits are set at load time, others (including some of those set at load time) may be changed during program execution. Two of the bits are of interest in this discussion, the others are documented in the RSTS/E System Directives Manual.

Bit 14 of the JSW controls automatic conversion of lower-case terminal input. When bit 14 is clear, lower-case characters typed in are automatically converted to upper-case. When bit 14 is set, no conversion is done. This bit is cleared at load time but may be set or cleared by the user program at any point during execution. Figure 2 presents an assembly language subroutine which when called from a FORTRAN program will enable lower-case input from the terminal by setting bit 14 of the JSW. In order to call this routine from a FORTRAN program you would put a "CALL LCENBL" into the code. The ".GLOBL LCENBL" directive sets LCENBL up as a global symbol so that it may be referenced from outside this program unit.

In order to disable lower-case input to the terminal only two things need to be changed in the routine. All occurrences of "LCENBL" should be changed to "DSABLC" and the "BIC" (bit clear) instruction should be substituted for the "BIS" (bit set) instruction. The new routine should then be referenced with a "CALL DSABLC" as before. Thus with these two assembler subroutines in your repertoire you may enable and disable lower-case terminal input as often as desired in your FORTRAN programs.



Pound for Pound, Stone for Stone or Kilogram for Kilogram The IBC Ensign™ Out Performs Them All.

68000 based computers are springing up all over the world. But, if you need a 16 bit microcomputer with performance to rival even the largest minicomputers, then you need the IBC Ensign, it out performs them all.

The Ensign is a high performance 16 bit microcomputer employing the most advanced VLSI techniques available along with the ultimate in expandability. The system features support for up to 32 users; up to 8M bytes of proprietary double bit ECC memory; over 1,000M bytes of SMD disk capacity; both cartridge and 9 track magnetic tape support; and operating system support for UNIX™ and OASIS-16™

The above features along with our exclusive multi-slave microprocessor architecture, make the Ensign a super microcomputer. The system utilizes two slave micro's with a 16K byte buffer to handle all I/O. A third slave controls all disk and tape I/O. Another micro provides memory management with capabilities and speed significantly beyond industry standard MMU circuits. This leaves the 68000 CPU free to process at its full instruction speed without any wait states. The result is a microcomputer that rivals the largest minicomputers in performance and expandability.

OEM's system integators and dealers! To find out more about the IBC microcomputer that outperforms them all and the best discount schedule and dealer plan in the industry, please call or write:

OUTSIDE THE USA

IBC/ Integrated Business Computers

21592 Marilla Street
Chatsworth, CA 91311
(213) 882-9007 TELEX NO. 215349

WITHIN THE USA

IBC/ DISTRIBUTION

1140 36th Street, Suite 212
Ogden, UTAH 84403
(801) 621-2294

UNIX is a trademark of Bell Laboratories
OASIS is a trademark of Phase One Systems

DEALERS CIRCLE D113 ON READER CARD
END USERS CIRCLE D151 ON READER CARD

```
;*****************************************************************
;*-------------- General  Comments ----------------------------*
;*****************************************************************
;
;
;       This routine is used to enable a FORTRAN program to accept lower
;       case ASCII characters from the user's terminal. It was written
;       specifically for the RT11 RTS under RSTS/E  V07.1.
;
;   James W. Sewall Company
;   Department of Forestry and Natural Resources
;   Old Town, Maine   04468
;
;   The information in this software module is subject to change
;   without notice and should not be construed as a commitment by
;   the James W. Sewall Company. Furthermore, the James W. Sewall
;   Company assumes no liability whatsoever in connection with the
;   use of this software module.
;
;       15-Oct-82                               Jeffrey H. Gove
;
;*****************************************************************
;*-------------- Assembler  Directives ------------------------*
;*****************************************************************
;
;
        .LIST   ME,TTM
        .TITLE  LCENBL
        .IDENT  /V01A/
        .GLOBL  LCENBL
;
;
;*****************************************************************
;*-------------- Begin  the  Routine --------------------------*
;*****************************************************************
;
;
LCENBL: BIS     #40000,@#44             ;set bit 14 of the JSW
        RTS     PC
        .END    LCENBL
```

FIGURE 2. LCENBL.MAC — Enabling Lower-case

The other bit of interest in the JSW is bit 12. This is the "special mode terminal bit." If bit 12 is cleared, terminal I/O proceeds as usual; characters are only available to the user program when an end-of-line is encountered. In addition, all characters are echoed back to the screen (including carriage return and line feed characters). This is the default setting at load time. When bit 12 is set, however, the program is in "ODT-mode." Characters are immediately available to the user's program as they are typed in and must be echoed back to the terminal by the program. Carriage return and line feed characters must be read in and checked for by the user's program. Control-O and control-C are processed as usual in "ODT-mode." Figure 3 presents an assembly language subroutine which will enable "ODT-mode" I/O at the user's terminal. This routine is accessed from a FORTRAN program by coding a "CALL ODTON" statement in the calling program.

As with bit 14, bit 12 may be set and cleared as many times as is desired in the user's program. It will be left as an exercise for the reader to determine how to turn off "ODT-mode."

Subroutine Linkages Under RT11

Both FORTRAN and MACRO use the instruction "JSR PC,name" to pass control to a subroutine or function subprogram. In FORTRAN it might appear a little differently,

```
;*****************************************************************
;*-------------- General  Comments ----------------------------*
;*****************************************************************
;
;
;       This routine is used to put a user's terminal into "ODT" mode.
;       It should be used only for programs running under
;       the RT11 run-time system on RSTS/E as it sets bit 12 in the
;       JSW  =  absolute address 44 of the user's job image.
;
;   James W. Sewall Company
;   Department of Forestry and Natural Resources
;   Old Town, Maine   04468
;
;   The information in this software module is subject to change
;   without notice and should not be construed as a commitment by
;   the James W. Sewall Company. Furthermore, the James W. Sewall
;   Company assumes no liability whatsoever in connection with the
;   use of this software module.
;
;       18-Nov-82                               Jeffrey H. Gove
;
;*****************************************************************
;*-------------- Assembler  Directives ------------------------*
;*****************************************************************
;
;
        .LIST   ME,TTM
        .TITLE  ODTON
        .IDENT  /V01A/
        .GLOBL  ODTON
;
;
;*****************************************************************
;*-------------- Begin  the  Routine --------------------------*
;*****************************************************************
;
;
ODTON:  BIS     #10000,@#44             ;set bit 12 in the JSW
        RTS     PC
        .END    ODTON
```

FIGURE 3. ODTON.MAC — Setting "ODT-mode"

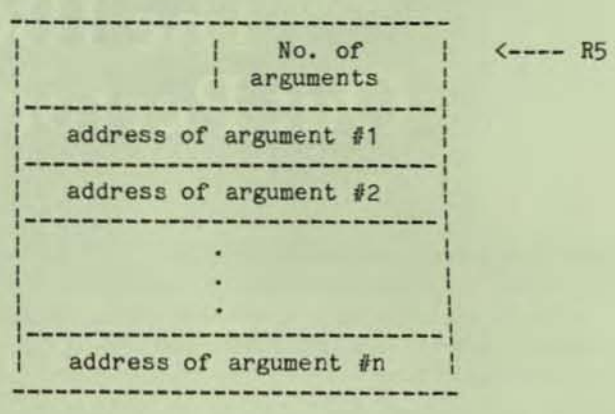
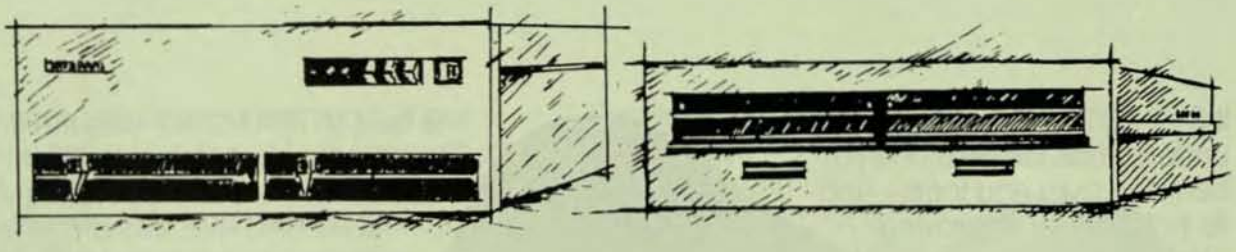


FIGURE 4. Subprogram Argument Block

"CALL name(A,B)" but it gets translated into the above instruction.

In the assembly language subroutines presented thus far no parameters have been passed. Figure 4 shows how parameters are passed to subroutines and function subprograms. Register 5 is set up to point to the address of the number of arguments in the list. This is the first byte in the subprogram argument block which is n+1 words long (where n is the number of arguments). Stepping through the argument block is most easily done with some form of the autoincrement addressing mode which is coded as "(R5)+" in assembly language. If an argument is missing in the parameter list (coded as (A,,B) in FORTRAN), a negative one is stored in the argument block in place of the address to that argument. It is important to note here that when FOR-

DEC Compatibility... with 8" RX02!



Why be forced to step down to 5¼" floppies on DEC's newer systems when you know you really want to maintain media compatibility with your existing 8" RX02-based systems? Dataram agrees, and offers an innovative series of system options that provide 8" RX02 capability for your PDP-11, LSI-11, or VAX minicomputers.

Q-bus and UNIBUS FD-311 subsystems provide dual RX-02-compatible 8" floppy drives for $2490. Or our dual RX02-compatible 8" slimline floppies, combined with an 8-quad slot Q-bus card cage, LSI-11/23, and 256KB memory are available for only $6510. Both products are supported by the industry's widest range of LSI-11 compatible products. Call or write for details.

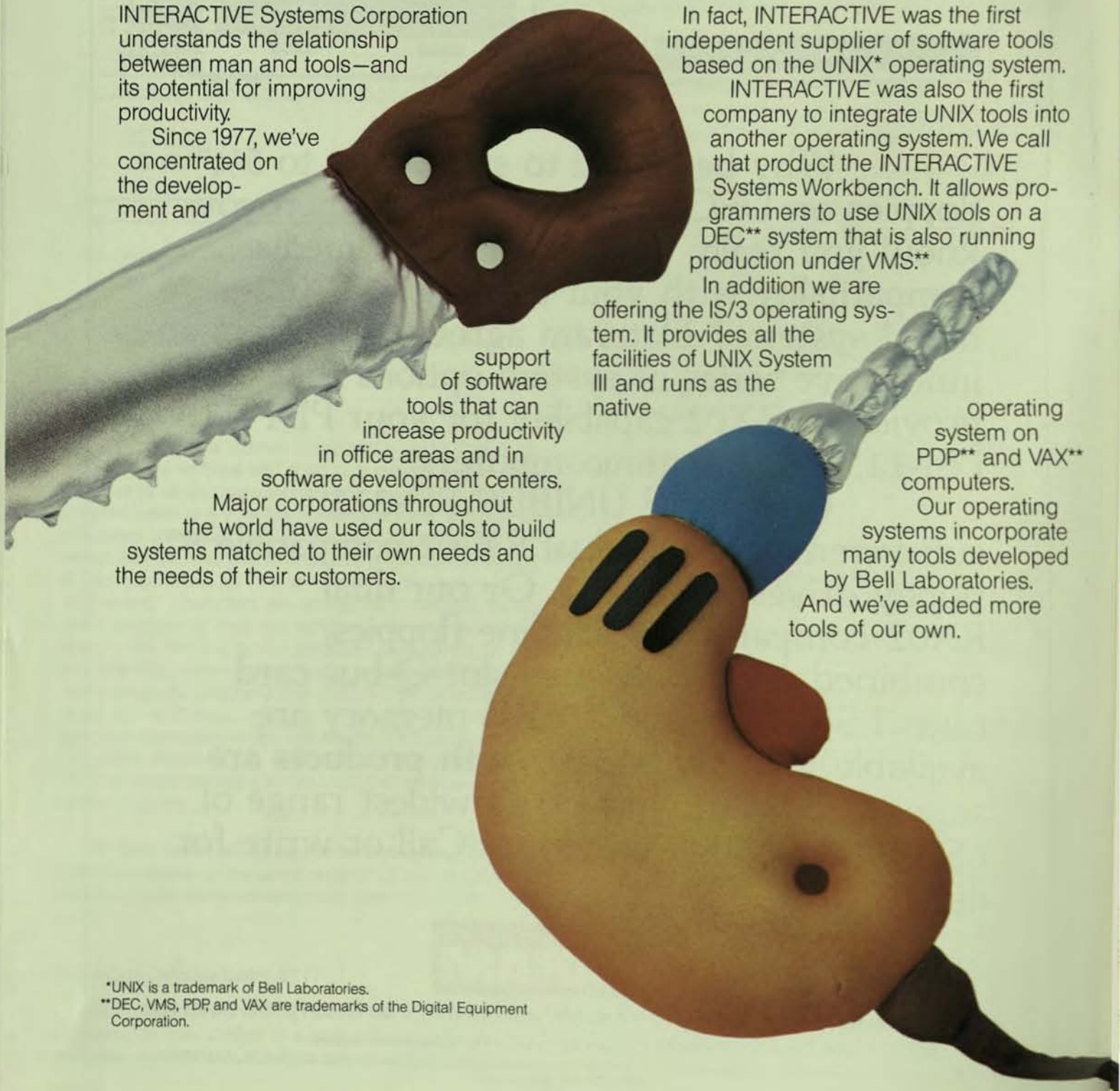
DATARAM

Dataram Corporation □ Princeton Road □ Cranbury, New Jersey 08512 □ Tel: 609-799-0071 □ TWX: 510-685-2542

DEC, LSI-11, PDP, UNIBUS and VAX are trademarks of Digital Equipment Corporation.

CIRCLE D116 ON READER CARD

MAN IS A TOOL-MAKING A
HE IS NOTHING. WITH TO



INTERACTIVE Systems Corporation understands the relationship between man and tools—and its potential for improving productivity.

Since 1977, we've concentrated on the development and support of software tools that can increase productivity in office areas and in software development centers.

Major corporations throughout the world have used our tools to build systems matched to their own needs and the needs of their customers.

In fact, INTERACTIVE was the first independent supplier of software tools based on the UNIX* operating system.

INTERACTIVE was also the first company to integrate UNIX tools into another operating system. We call that product the INTERACTIVE Systems Workbench. It allows programmers to use UNIX tools on a DEC** system that is also running production under VMS.**

In addition we are offering the IS/3 operating system. It provides all the facilities of UNIX System III and runs as the native operating system on PDP** and VAX** computers.

Our operating systems incorporate many tools developed by Bell Laboratories. And we've added more tools of our own.

*UNIX is a trademark of Bell Laboratories.
**DEC, VMS, PDP and VAX are trademarks of the Digital Equipment Corporation.

NIMAL. WITHOUT TOOLS, OLS, HE IS EVERYTHING.

—From Thomas Carlyle, 1795-1881

One of the most important tools is our full-screen text editor. It uses features such as multiple windows, a HELP key and function keys to significantly improve the user interface to UNIX.

The editor is a good example of the way we have enhanced UNIX. Our editor is easier to learn and easier to use than the standard UNIX editors. It can make your host computer more efficient by allowing you to move certain tasks from your host computer to intelligent terminals. By using that feature, you can provide better response times and support more users.

We're now offering still another innovation. We have packaged sets of our tools to meet the most common customer needs and are offering the sets at new low prices. All of these packages run under either of our operating systems.

INTERACTIVE's Advanced Productivity System is our most complete set. It gives you a packaged set of tools to enhance both programming and office productivity. APS includes one of our operating systems, our screen editor, a versatile electronic mail facility, a networking capability, word-processing enhancements and two of our intelligent terminals. INTERACTIVE also provides free installation and training plus a year of free support and maintenance.

Yes, we've accepted the wisdom of Thomas Carlyle. We're committed to building tools that our customers can use "to be everything."

If you need such tools, call us at 213 - 450-8363. Or write,

INTERACTIVE Systems Corporation
1212 Seventh Street
Santa Monica, CA 90401.

INTERACTIVE
SYSTEMS CORPORATION

CIRCLE D123 ON READER CARD

FORTRAN . . .

TRAN is calling an assembly language routine all of the hardware registers are preserved on the system stack before the call is made.

FORTRAN uses the same method of passing arguments to other FORTRAN subprograms via "dummy" variables but it is camouflaged to the user. Thus it is enlightening to see how a FORTRAN to assembly language subprogram linkage works, in order to better understand exactly what you are coding in a FORTRAN to FORTRAN linkage.

An example of an assembly language subroutine which needs parameters passed from the calling routine is shown in Figure 5. This routine is used to erase all or part of the screen on a VT100 terminal. Documentation as to how this routine is used is included under the "General Comments" section in the listing.

Several lines in ERS100 are worth mentioning in more detail. Line *1* shows the use of the .MCALL directive. This statement must be included in the code with a list of the RT11 emulator directives which will be used in the program prior to their use. In this case .TTYOUT is the only directive to be used in ERS100. The macro expansions of .TTYOUT and other RT11 emulator directives are found in the system macro library (see Assembling and Linking).

Line *2* simply skips over the number of arguments in the argument block and positions R5 so that it is now pointing to the address of the first argument. Lines *3* and *4* move the contents of the variables ICHOIC and IPARAM into registers R4 and R1 respectively.

Line *5* shows how to send a byte to the terminal with the .TTYOUT directive. In addition, for anyone who has had trouble determining how to send an escape sequence to a VT100, it is interesting to note that RSTS changes escape characters ("33) to dollar sign characters ("44) when they are being sent to the terminal. To get around this and initiate a valid VT100 escape sequence, "233 must be sent to the terminal.

Line *6* shows how to call another assembly language subroutine (ASCOUT is listed in Figure 8 in the appendix). Finally, line *7* shows how a return from a subroutine to a calling program is accomplished in both FORTRAN and assembly language ("RETURN" replaces "RTS PC" in FORTRAN).

Function subprograms are coded similarly to subroutines in assembly language. Argument lists are passed identically, however, a single variable result is returned in one of the hardware registers depending upon the type of function as shown in table 1.

```
;*************************************************************
;*------------ G e n e r a l   C o m m e n t s ------------*
;*************************************************************
;
;
;   This routine allows the user of the VT100 terminal to do screen
;   or line erasures. The routine may be called from a FORTRAN
;   program as follows...
;
;           CALL ERS100(ICHOIC,IPARAM)
;
;   Where:  ICHOIC  =       1 = erase in display (ED)
;                           2 = erase in line (EL)
;
;           IPARAM  =       depends on ICHOIC as follows...
;
;     ICHOIC   IPARAM             Explanation
;   ------------------------------------------------------
;
;           1       0 (default)     Erase from active position to
;                                   the end of screen, inclusive
;                   1               Erase from start of screen to
;                                   active position, inclusive
;                   2               Erase all of display - all lines
;                                   are erased, changed to single-width
;                                   and the cursor does not move
;
;           2       0 (default)     Erase from active position to the
;                                   end of line, inclusive
;                   1               Erase from start of line to the
;                                   active position, inclusive
;                   2               Erase all of the line
;
;   If anything other than a 1 or 2 is passed for ICHOIC -- the routine
;   exits and does nothing. If anything other than a 0, 1 or 2 is
;   passed for IPARAM -- the default option is always used for the
;   ICHOIC chosen.
;
;   Thus if it is desired to erase the entire screen regardless of
;   where the current cursor position is, you would code the following
;   call in FORTRAN...
;
;                   CALL ERS100(1,2)
;
;
;   James W. Sewall Company
;   Department of Forestry and Natural Resources
;   Old Town, Maine   04468
;
;   The information in this software module is subject to change
;   without notice and should not be construed as a commitment by
;   the James W. Sewall Company. Furthermore, the James W. Sewall
;   Company assumes no liability whatsoever in connection with the
;   use of this software module.
;
;   12-May-83                               Jeffrey H. Gove
;
;*************************************************************
;*------------ A s s e m b l e r   D i r e c t i v e s ------*
;*************************************************************
;
;   written in ... MACRO-11 V04.0 under RSTS/E V07.1
;
        .LIST   ME,TTM
        .TITLE  ERS100
        .IDENT  /V01A/
*1*     .MCALL  .TTYOUT
        .GLOBL  ERS100,ASCOUT
;
;*************************************************************
;*------------ B e g i n   t h e   R o u t i n e ----------*
;*************************************************************
;
*2* ERS100: TST     (R5)+                   ;skip the # of arguments
*3*         MOV     @(R5)+,R4               ;get the erasure choice
*4*         MOV     @(R5),R1                ;get the parameter into R1
;
;   check the parameter first...
;
            CMP     #1,R1                   ;is it a 1?
            BEQ     GO                      ;go on if so
            CMP     #2,R1                   ;is it a 2?
            BEQ     GO                      ;go on if so
            CLR     R1                      ;0 is the default
;
;   output the escape sequence and parameter...
;
*5* GO:     .TTYOUT #233                    ;esc
            .TTYOUT #133                    ;[
*6*         JSR     PC,ASCOUT               ;send the parameter
;
;   check for the type of erasure...
;
            CMP     #1,R4                   ;screen erase?
            BEQ     SCREEN                  ;yes
            CMP     #2,R4                   ;line erase?
            BNE     GOBACK                  ;bad choice passed--exit
            .TTYOUT #113                    ;K -- erase in line
            BR      GOBACK                  ;done
    SCREEN: .TTYOUT #112                    ;J -- erase in display
*7* GOBACK: RTS     PC
            .END
```

FIGURE 5. ERS100.MAC — Erase a VT100 Screen

The Long Wait Is Over

The CSS-800 Compact Storage System

Work loads are growing, you need more storage, your system can't keep up the pace, and everyone's waiting for you to come up with an answer. The CSS-800 is the answer you've been waiting for—a storage system with larger capacity and data management features needed to upgrade system performance.

When long waits for disk access hold back your system and limit its throughput, faster storage is the key to performance. The CSS-800's exclusive architecture with cache memory and look-ahead buffering can speed throughput up to 90%. Where other systems control data flow simplistically, the 800 strategically responds to software and data usage without adding to host overhead time. With 70 Mbytes of disk storage, your mini or micro can store more data, handle more concurrent users, run large programs more effectively, and do more processing in less time than ever before.

When More Is Less Our 70-Mbyte disk and integral cartridge tape backup are housed in a rackmount or desktop package 5¼ inches high and weighs only 42 pounds. Each controller can accommodate three add-on storage systems for a total on-line capacity of 280 Mbytes. When space is as important as time, you can't beat the CSS-800.

When More Costs Less The CSS-800 is priced below any other system with comparable capacity to give you the best cost per Mbyte and price/performance ratios in the industry. The 800 gives you more capacity and greater speed for less money than you'd spend for an ordinary disk drive.

On DEC Now! Totally DEC* compatible, the CSS-800 emulates the RK07 disk and TU10 tape drives so there is DEC system support available now under RSX-11M, RSTS/E, and RT-11 operating system as well as DSM, TSX and UNIX. Q-bus or Unibus host interfaces are standard features.

And Don't Wait for Repairs We also know that you need a storage unit that is dependable as well as fast. The CSS-800 uses Winchester technology to achieve maximum availability. Built-in diagnostics assure the reliability of all drives, controllers, and interfaces. Modular design makes repairs fast and easy. We've backed the 800 with TRW's nationwide service program to insure prompt and complete maintenance.

U.S. Design Corporation is a leader in providing performance enhancements for the mini/micro computer community. Call us toll free at (800) 368-2811 TODAY and we'll show you the technology of TOMORROW.

*DEC, RK07, TU10, RSX-11, RSTS/E, RT-11, DSM, Q-Bus and Unibus are registered trademarks of Digital Equipment Corporation. TSX is a registered trademark of S & H Computers. UNIX is a registered trademark of Bell Laboratories.

U.S. DESIGN CORPORATION
5100 Philadelphia Way
Lanham, Maryland 20706
(301) 577-2880 (800) 368-2811
TWX 710-828-0417

USDC

CIRCLE D118 ON READER CARD

TABLE 1. Registers Used For Function Subprogram Results

Function Type	Result Placed in Register...
LOGICAL*1 INTEGER*2	R0
LOGICAL*4 INTEGER*4	R0 = low-order result R1 = high-order result
REAL*4	R0 = high-order result (including sign and exponent) R1 = low-order result
REAL*8	R0 = high-order result (including sign and exponent) R1 = next highest-order result R2 = next highest-order result R3 = lowest-order result
COMPLEX	R0 = high-order real result R1 = low-order real result R2 = high-order imaginary result R3 = low-order imaginary result

An example of an INTEGER*2 function written in assembly language is presented in Figure 6. The "General Comments" section outlines how the function might be called from a FORTRAN program. Note that R0 is used as the character counter in order to determine where in the string the character being searched for is located. If the character is found then its position in the string is returned in R0, otherwise R0 is cleared and FNDCHR returns the result of zero. FNDCHR must be explicitly declared as an INTEGER*2 function in the FORTRAN calling program, otherwise it will default to a REAL function.

One last point of interest about subprogram linkages is that the number of arguments in the argument list passed may vary when calling an assembly language routine from FORTRAN. Referring back to Figure 4, R5 is set pointing to the number of arguments in the argument list when the call is made. This may be used by the assembly language program (rather than skipped over as in my examples) to set up a loop counter which will access the values of the arguments in the list and perform some type of function on them. For example, "CALL SUBA(A,B,C)" and "CALL SUBA(A,B,C,D,E)" could both be handled very neatly in the assembly language subprogram SUBA by using the number of arguments to access the correct number of arguments in the list each time the call is made. No provision is made for this FORTRAM to FORTRAN calls where the actual arguments must agree in order, number and variable type with the corresponding dummy arguments of the referenced subprogram.

Using RSTS/E General Monitor Directives

Perhaps the most powerful argument for using assembler language routines in order to augment FORTRAN is the ability to use almost all of the general monitor directives supported by RSTS/E. These directives are where BASIC-PLUS SYS calls have their roots. Although DEC left it up to FORTRAN programmers to take it upon themselves to write any routines needed, it is worth the extra time to actually do this in order to have access to this powerful set of tools.

In addition to the general monitor directives, the RT11 Emulator allows the use of many of the RT11 Operating System monitor calls. The .TTYOUT routine used in ERS100 is one such directive. Additionally, several RSTS/E-specific

```
;*****************************************************************
;*---------------  G e n e r a l   C o m m e n t s  ---------------*
;*****************************************************************
;
;       This routine is used specifically to look for a CHARACTER
;       in an ASCII string and return the location of the character
;       in the string. If the character is not found a value of zero
;       is returned. It should be used as a FORTRAN INTEGER*2
;       function as follows...
;
;               INTEGER         FNDCHR
;               LOGICAL*1       STRING(31),CHAR
;               DATA            STRING/31*0/            !null the string
;                       .
;                       .
;                       .
;               CHAR = '['
;               IPLACE = FNDCHR(STRING,CHAR)            !look for a "["
;               JPLACE = FNDCHR(STRING,'D')             !look for a "D"
;               KPLACE = FNDCHR(STRING,*62)             !look for a "2"
;                       .
;                       .
;                       .
;               STOP
;               END
;
;       where:  IPLACE = the byte in the string where the character
;                        was found. It is 0 if the character was not found in
;                        the  string.
;               STRING = an ASCII string with a null byte terminator.
;               CHAR   = an ASCII character
;
;   James W. Sewall Company
;   Department of Forestry and Natural Resources
;   Old Town, Maine   04468
;
;   The information in this software module is subject to change
;   without notice and should not be construed as a commitment by
;   the James W. Sewall Company. Furthermore, the James W. Sewall
;   Company assumes no liability whatsoever in connection with the
;   use of this software module.
;
;       27-May-83                                       Jeffrey H. Gove
;
;*****************************************************************
;*---------------  A s s e m b l e r   D i r e c t i v e s  -------*
;*****************************************************************
;
;       written in MACRO-11 V04.0 under RSTS/E V07.1
;
        .LIST   ME,TTM
        .TITLE  FNDCHR
        .IDENT  /V01A/
        .GLOBL  FNDCHR
;
;*****************************************************************
;*---------------  B e g i n   t h e   R o u t i n e  -------------*
;*****************************************************************
;
FNDCHR: TST     (R5)+                   ;skip the # of arguments
        MOV     (R5)+,R1                ;get string start address
        MOV     (R5),R2                 ;get the byte address
        CLR     R0                      ;R0 is the where the value
                                        ;will be returned via this
                                        ;integer function
10$:    CMPB    (R2),(R1)               ;is it the character?
        BEQ     ALMOST                  ;if so exit...
        TSTB    (R1)+                   ;are we at the end of string?
        BEQ     NOGO                    ;quit if so
        INC     R0                      ;increment string counter
        BR      10$                     ;back for another byte
;
;       almost done...we're 1 behind in the string pointer...
;
ALMOST: INC     R0                      ;so catch up
        BR      DONE                    ;exit character found
;
;       exit with the character not found...
;
NOGO:   CLR     R0                      ;counter set to zero
DONE:   RTS     PC
        .END
```

FIGURE 6. FNDCHR.MAC — Finds a Character In a String

WHAT MADE OUR DH/DM THE STANDARD REPLACEMENT, ALSO MADE IT THE STANDARD DOWN UNDER.

Because Able's DH/DM communications controller does so much more than the DEC DH11 and DM11 controllers, it's become the standard replacement in PDP-11 computers.

Except down under, in Australia. There, Able's DH/DM is not the standard replacement. There, it is the standard. In Australia, our DH/DM is standard equipment on PDP-11 computers, and on VAX computers running on UNIX.

It's easy to see the reasons our 16 line hex-size communications controller has become the standard.

Clearly, there's more space. The DH/DM plugs into one hex slot, instead of nine. That frees up eight slots for peripherals.

The DH/DM is faster. It transfers data by word rather than by byte. Plus, there's an input silo with four times the standard depth for even higher data transfer rates without ripple down delays or overflow. And with one-third less system loading and half the power consumption.

Modem control on all sixteen lines provides unobstructed system flexibility. And it's software compatible with DEC's DH11 and DM11.

To minimize obstacles, the DH/DM's self-diagnostics and LED display make it easy to locate and isolate problems.

And if you need them, Able's fast, professional service and support are as close as the phone. We're organized to solve problems in hours, not days. We want to provide you with complete customer satisfaction, and we work hard to do so.

Able has a broad range of devices for PDP-11 computers offering improved communications, memory expansion and interprocessor connectivity. All of which comply with FCC regulations.

So you see, you don't have to be down under to come out on top. Just get the controller that's become the standard, the DH/DM, from the people who introduced it.

ABLE COMPUTER

The communications specialists.

1732 Reynolds Avenue, Irvine, California 92714. Call toll free: 800-332-2253. In the Irvine area: (714) 979-7030. Or, TWX: 910-595-1729

DEC, VAX, DMF32 and VMS are trademarks of Digital Equipment Corporation.

CIRCLE D120 ON READER CARD

RT11 directives have been added. These directives are all documented in Chapter 7 of the RSTS/E System Directives Manual.

Presented in Figure 7 is an assembly language program which does a directory lookup. This subprogram is used to first see if the file specification string passed by the calling program is legal for RSTS/E and subsequently to see if the file resides on the device requested in the file specification string.

Again, several lines are worth mentioning in more detail here. Line *1* illustrates how to define the RSTS/E-specific RT11 directive .CLRFQB for use in the routine. This directive is then actually used in line *2* to clear the File Request Queue Block (see Courtney, (1981) and Chapters 2 and 3 of the RSTS/E System Directives Manual for discussions of FIRQB, XRB and their associated mnemonics). Note that the same procedure is used to clear the Transfer Control Block (XRB) using the .CLRXRB directive. A file string scan examines the file specification string and places information about the string into the FIRQB and XRB via the .DOFSS directive in line *3*. The same directive returns information on the validity of the string in the XRB. The XRB can then be checked to see if a legal file specification string had been passed.

The CALFIP (Call the File Processor) function code for a directory lookup is moved into the correct part of the FIRQB in line *4*. Line *5* shows the "special prefix" EMT which must precede any attempt to use one of the RSTS/E general monitor directives. Finally, in line *6* the file processor is called and the directory lookup is accomplished. The rest of the routine simply determines what value IERR will take on when control is returned to the calling program.

A Word About MACRO Assembly and Linking

The one last key to FORTRAN — MACRO interfacing is to be able to assemble your MACRO-11 subprograms and link them with your FORTRAN programs. This is accomplished by creating an object file from your assembly language program with the assembler and then linking that object file with others whether they be from FORTRAN or MACRO.

The MACRO assembler is used in order to create an object file from an assembler source file. To assemble the program ERS100 in Figure 5, the following command would be typed in . . .

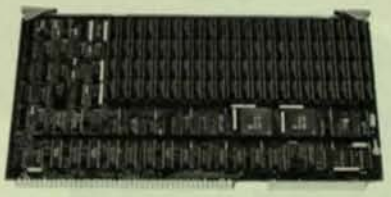
```
MACRO ERS100,ERS100=ERS100
```

This command will create both an object and a list file just as the FORTRAN compiler does. In order to correctly assemble ERS100 and other assembly language routines that use the RT11 emulator directives, the file $SYSMAC.SML (System Macro Library File) distributed with RSTS/E must reside on the system.

FILOOK.MAC (Figure 7) must be assembled a little differently since it relies on the file COMMON.MAC (also sup-

MEMORIES

512KB TO 2 MB EDC SINGLE BOARD MULTIBUS MEMORY



- Pin to Pin MULTIBUS compatibility for both 8 bit and 16 bit systems.
- Addressable up to 16 megabytes.
- On board parity with selectable interrupt on parity ERROR.

OR

- Error Detecting and Correcting (EDC).

SINGLE QTY. PRICE:

	Without/EDC	W/EDC
512KB	$ 895.00	$1495.00
2MB	$5995.00	$6495.00

NEW! NEW!

2MB Q-BUS EDC 512KB TO 2MB QUAD Q-BUS MEMORY

WITH ERROR DETECTION AND CORRECTION

UNBELIEVABLE LOW PRICE!

Call for details!

256KB TO 1 MB SINGLE BORAD LSI 11 MEMORY

- Control Status Register (CSR)
- On board parity generator checker.
- Battery back-up mode.

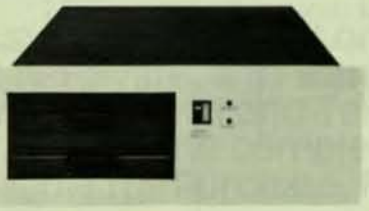
SINGLE QTY. PRICE:
256KB $525.00

DISK SYSTEMS COMPUTERS

2MB FLOPPY

Dual drive, double density, double sided 2MB floppy (8") with controller and bootstrap. Also reads single sided, single density!

CI-1220 TF $2695.00



42MB, 70MB, 140MB formatted Winchester (8") disk system w/controller and 2MB floppy (8") backup w/controller.

CI-1240-WF: 42MB Winchester $6995.00
CI-1270-WF: 70MB Winchester $7495.00
CI-1340-WF: 140MB Winchester $9995.00

10MB WINCHESTER OR 20MB WINCHESTER

RL02 compatible Winchester subsystems. 2MB floppy single dual controller and bootstrap is supplied with each purchase.

CI-820:
10MB Winchester 2MB floppy (8") $4495.00
20MB version add $500.00

CI-11/23AC: LSI 11/23CPU, 256KB, 4 x 8 backplane, power supply and chassis (excluding rack slides) **$2675.00**
CI-11/23AF: LSI 11/23 + CPU, 1MB, 4 x 8 backplane, power supply and chassis (excluding rack slides) **$4125.00**

"OFFERING QUALITY WITH AFFORDABLE PRICING"

Chrislin Industries, Inc.

31352 Via Colinas • Westlake Village, CA 91362
Telephone: 213-991-2254 • TWX 910-494-1253 CHRISLIN WKVG

Multibus is a Trademark of Intel Corporation. QBUS, LSI, is a Trademark of Digital Equipment Corporation.

CIRCLE D153 ON READER CARD

plied with RSTS/E) to define many of the mnemonics used such as FIRQB. To correctly assemble FILOOK the following command would be used . . .

MACRO FILOOK=COMMON,FILOOK

This time only an object file is created but there is no reason why a list file could not be generated as well if desired.

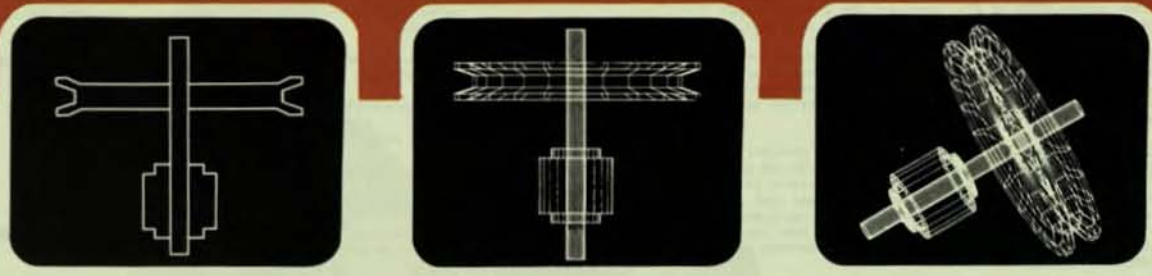
```
;*********************************************************************
;*------------ General  Comments ----------------------*
;*********************************************************************
;
;
;   This routine is used to do directory lookups from a FORTRAN
;   calling program. By calling this routine, the FORTRAN program
;   can determine...
;
;           (1)     if the file specification entered by the
;                   user is a legal RSTS/E file string
;
;           (2)     if the file in fact exists on a specified
;                   directory or not
;
;   The following examples illustrate legal calls to this routine...
;
;           LOGICAL*1       NAM(31)
;           DATA    NAM/31*0/       !null the string
;   10      TYPE *,'enter the file name...'
;           ACCEPT 20,LEN,(NAM(I),I=1,LEN)
;   20      FORMAT(Q,80A1)
;           CALL FILOOK(IERR,NAM)
;           IERR = IERR + 1
;   C
;   C       what has happened?...
;   C
;           GOTO (100,200,300),IERR
;   C
;   100     TYPE *,'File found!'
;           GOTO 350
;   C
;   200     TYPE *,'Bad file name!'
;           GOTO 350
;   C
;   300     TYPE *,'File not found!'
;   350     DO 400 I=1,31
;   400     NAM(I)=0
;           GOTO 10
;           STOP
;           END
;
;   As this example illustrates, the determination as to whether
;   the file exists etc. is made by testing the value of IERR
;   as returned by FILOOK...
;
;           IERR = 0 ...    the file was found
;                = 1 ...    bad file string specification
;                = 2 ...    the file does not exist
;
;   An alternative way of calling FILOOK is shown below...
;
;           CALL FILOOK(IERR,'#CNTRL.DAT')
;
;   Thus the system !, @, #, $, % and & account specifiers may be
;   included as part of the file string specification and the string
;   may also be passed as a literal as shown.
;
;
;   James W. Sewall Company
;   Department of Forestry and Natural Resources
;   Old Town, Maine  04468
;
;   The information in this software module is subject to change
;   without notice and should not be construed as a commitment by
;   the James W. Sewall Company. Furthermore, the James W. Sewall
;   Company assumes no liability whatsoever in connection with the
;   use of this software module.
;
;
;   10-May-83                               Jeffrey H. Gove
;
;
;*********************************************************************
;*------------ Assembler  Directives ------------------*
;*********************************************************************
;
;
        .LIST   ME,TTM
        .TITLE  FILOOK
        .IDENT  /V01A/
        .GLOBL  FILOOK
        .PSECT                          ;this must be here because of
                                        ;COMMON.MAC's use of .RSECT
;
;
;*********************************************************************
;*------------ Begin  the  Routine --------------------*
;*********************************************************************
;
;
;   Define the RT-11 Emulator directives used...
;
*1* .CLRFQB =       EMT+360             ;clears FIRQB through RT-11
    .DOFSS  =       EMT+365             ;does FSS through RT-11
    .CLRXRB =       EMT+371             ;clears XRB through RT-11
;
;       and begin...
;
    FILOOK: TST     (R5)+               ;skip the # of arguments
            MOV     R0,-(SP)            ;save the contents of R0
            MOV     R3,-(SP)            ;ditto for R3
            MOV     (R5),R3             ;get the address of error code
            MOV     (R5),R0             ;get string address into R0 --
                                        ;DOFSS needs it there
*2*         .CLRFQB                     ;clear the FIRQB
            .CLRXRB                     ;clear the XRB
*3*         .DOFSS                      ;do a FSS-loads stuff into FIRQB
            BIT     #1,XRB+10           ;was a legal file spec found?
            BEQ     BADSTR              ;if not -- error
*4*         MOVB    #LOKFQ,FIRQB+FQFUN  ;set CALFIP function code
*5*         EMT     "0377               ;"special prefix" for monitor
                                        ;intercept of CALFIP...
*6*         CALFIP                      ;call the file processor
            TST     FIRQB+24            ;file found? -- check creation
                                        ;date to be sure...
            BEQ     NOFILE              ;no file found -- set error flag
;
;       file was found ... set no error in the flag...
;
            CLR     (R3)                ;0 = file was found
            BR      FIXUP
;
;       bad string specification by the user detected in FSS...
;
    BADSTR: MOV     #1,(R3)             ;1 = bad string specs
            BR      FIXUP
;
;       file was not found ... set an error...
;
    NOFILE: MOV     #2,(R3)             ;2 = file was not found
;
;       fix up the stack and exit...
;
    FIXUP:  MOV     (SP)+,R3            ;restore contents of R3
            MOV     (SP)+,R0            ;restore contents of R0
            RTS     PC
            .END
```

FIGURE 7. FILOOK.MAC — Directory Lookup Subroutine

HAMILTON

CADD System

Computer Aided Design and Drafting



For as little as $25,000, HAMILTON provides a complete hardware-software CADD System.

At the heart of this package is Design Graphix,™ a full-function 3D CADD System, which includes TEACH, an on-line programmed course providing a thorough understanding and mastery of the system in just a few days.

This dynamic system is layered on DEC PRO 350, PDP-11 or VAX hardware, and supports a full range of Hewlett Packard and Tektronix graphics peripherals. The complete CADD System is available for Purchase, Rental or Time Sharing.

For further information, demonstration or individual quotation, call or write NOW!

HAMILTON **THE HOUSE OF VAX**

HGL Software Hamilton Rentals

6 Pearl Court, Allendale, NJ 07401
TOLL FREE 800-631-0298
In New Jersey 201-327-1444

415 Horner Ave., Toronto M8W4W3
TOLL FREE Ont. & Que. 800-268-2106
All other prov. 800-268-0317
In Canada 416-251-1166

NEW YORK • DALLAS • MONTREAL • CALGARY • LONDON • PARIS • DUSSELDORF

APPENDIX

```
;*****************************************************************
;*-------------- General  Comments ---------------------------*
;*****************************************************************
;
;
;       This routine is used specifically to output a row or column
;       coordinate to the VT100 or VT52 terminal. One integer number
;       no larger than 99 may be passed to this routine at a time.
;
;       If a number greater than two digits is passed a fatal error
;       occurs in ASCOUT and this routine will halt the entire program.
;       Thus it would be wise to check for legal screen addresses in
;       the calling routine and handle any errors there before ASCOUT
;       gets the number.
;
;       This routine MUST be called by another assembler routine. It
;       expects to find the number which will be made ASCII and passed
;       to the terminal in R1. R0 and R1 are used for division in this
;       routine and R3 is used as a check on screen address size. Thus
;       the calling routine might want to preserve these registers on
;       the system stack before the call is made. At the very least
;       the calling routine must get the integer screen address into
;       R1 before the call is made!
;
;   James W. Sewall Company
;   Department of Forestry and Natural Resources
;   Old Town, Maine   04468
;
;   The information in this software module is subject to change
;   without notice and should not be construed as a commitment by
;   the James W. Sewall Company. Furthermore, the James W. Sewall
;   Company assumes no liability whatsoever in connection with the
;   use of this software module.
;
;       11-May-83                                   Jeffrey H. Gove
;
;
;*****************************************************************
;*-------------- Assembler  Directives -----------------------*
;*****************************************************************
;
;
        .LIST   ME,TTM
        .TITLE  ASCOUT
        .IDENT  /V01A/
        .MCALL  .TTYOUT,.EXIT,.PRINT
        .ENABL  LC
        .GLOBL  ASCOUT,SET100
;
;
;*****************************************************************
;*-------------- Memory / Storage ----------------------------*
;*****************************************************************
;
;
ENDLIN: .ASCIZ  <7><15><12>             ;bell ring and crlf first
TEXT:   .ASCIZ  /**>Error--Too large a screen address in ASCOUT!/
        .EVEN
MEM:    .WORD   0                       ;save a word for the number
;
;
;*****************************************************************
;*-------------- Begin  the  Routine -------------------------*
;*****************************************************************
;
;
ASCOUT: MOV     #MEM,R2                 ;define memory storage
        CLR     (R2)                    ;clear the word each call
        CLR     R3                      ;clear the digit counter
1$:     INC     R3                      ;increment the number check counter
        CMP     #3,R3                   ;too many digits?
        BEQ     TOOBIG                  ;if so=error!
        CLR     R0                      ;clear R0
        DIV     #12,R0                  ;divide by ten
        ADD     #60,R1                  ;form the first ASCII byte
        MOVB    R1,(R2)+                ;and get it into memory
        MOV     R0,R1                   ;keep the remainder
        BNE     1$                      ;go back for another if need be
;
;       now print the number out last byte first then first byte...
;
2$:     MOVB    -(R2),R0                ;now get it into R0
        .TTYOUT                         ;and print
        CMP     #MEM,R2                 ;are we done?
        BNE     2$                      ;nope if true!
        RTS     PC                      ;exit
;
;       this portion of the routine is used to print an error
;       if there are too many digits passed to it--FATAL!
;
TOOBIG: JSR     PC,SET100               ;clear the screen
        .PRINT  #ENDLIN
        .PRINT  #TEXT
        .EXIT
        .END
```

FIGURE 8. ASCOUT.MAC — ASCII Integer Output To TTY

Once the object files have been generated they are in the same format as FORTRAN generated object files. Thus you can put them into libraries for general use or link them directly with other FORTRAN and MACRO routines. Both the MACRO assembler and the Linker are described in detail in the RSTS/E RT11 Utilities Manual. Chapters 2 and 3 respectively.

```
;*****************************************************************
;*-------------- General  Comments ---------------------------*
;*****************************************************************
;
;       This routine is used to set the ANSII mode on a VT100 terminal.
;       After setting the ANSII mode it clears the screen and leaves the
;       cursor positioned in the location where it was when the call
;       was made.
;
;       This routine may be called from either another MACRO routine or
;       from a FORTRAN routine.
;
;       An example FORTRAN call follows...
;
;               CALL SET100
;
;       An example MACRO call follows...
;
;               JSR     PC,SET100
;
;       having declared the routine name as a global symbol with the
;       .GLOBL directive in the calling routine for MACRO.
;
;   James W. Sewall Company
;   Department of Forestry and Natural Resources
;   Old Town, Maine   04468
;
;   The information in this software module is subject to change
;   without notice and should not be construed as a commitment by
;   the James W. Sewall Company. Furthermore, the James W. Sewall
;   Company assumes no liability whatsoever in connection with the
;   use of this software module.
;
;       11-May-83                                   Jeffrey H. Gove
;
;
;*****************************************************************
;*-------------- Assembler  Directives -----------------------*
;*****************************************************************
;
;
        .LIST   ME,TTM
        .TITLE  SET100
        .IDENT  /V01A/
        .MCALL  .TTYOUT
        .GLOBL  SET100
;
;
;*****************************************************************
;*-------------- Begin  the  Routine -------------------------*
;*****************************************************************
;
;
;       first switch to ANSII mode...
;
SET100: .TTYOUT #233    ;esc
        .TTYOUT #74     ;<
;
;       and now erase the entire screen...
;
        .TTYOUT #233    ;esc
        .TTYOUT #133    ;[
        .TTYOUT #62     ;2
        .TTYOUT #112    ;J
        RTS     PC
        .END
```

FIGURE 9. SET100.MAC — Initialize a VT100 Screen

References

Courtney, Thomas. "Beginner's Guide to MACRO 11 Programming in RSTS/E." RSTS Professional 3, No. 1 (March 1981):26.

Eckhouse, Richard H. and Morris L. Robert. Minicomputer Systems Organization, Programming and Applications (PDP-11). 2nd ed. New Jersey: Prentice-Hall, 1979.

Gill, Arthur. Machine and Assembly Language Programming of the PDP-11. 1st ed. New Jersey: Prentice-Hall, 1978.

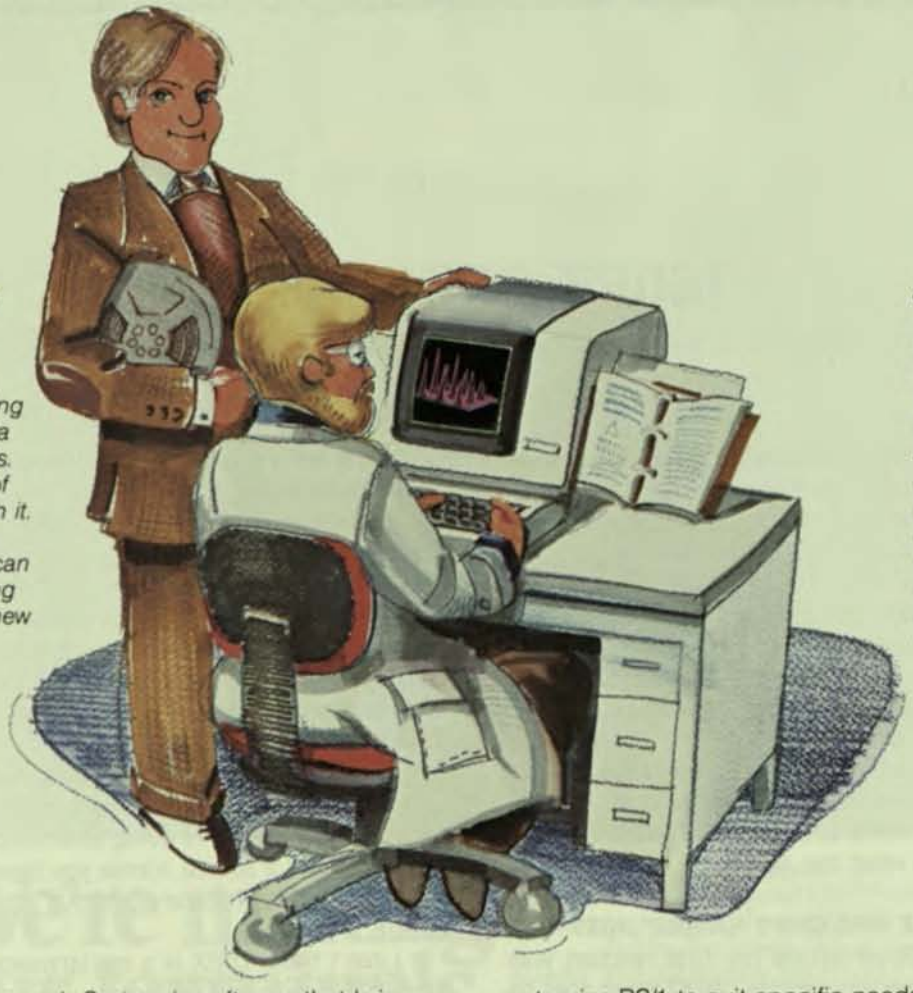
Singer, Michael. PDP-11 Assembler Language Programming and Machine Organization. 1st ed. New York: John Wiley and Sons, 1980.

RS/1™

The Research System

The Foundation of a Great Relationship

Data Processing Manager
I used to spend days-even weeks writing and rewriting applications to fit a researcher's needs. And neither one of us was happy with it. But now I have a package that he can use himself, freeing me up for major new systems.



Scientist
I used to wait forever to get my data out of DP. With RS/1 though, I have control. I can get statistics, graphics, and models easily by manipulating data myself, and I get it done right. RS/1 can help me find the breakthrough that's hidden in my data.

RS/1, The Research System is software that brings relief to the DP department. And results to scientists and engineers. A win/win situation.

With 150 person/years of development behind it, RS/1 is a mighty tool for research. Thousands of users—in hundreds of installations—already enjoy its speed and productivity.

RS/1 gives researchers the capability to do their own high-level data processing—without having to go through the DP department; and without having to write their own programs. RS/1 saves days, weeks, sometimes months, in working out problems.

RS/1 gives DP professionals an interactive program development environment with access to RS/1's high-level statistics, graphics, and DBMS so they can customize RS/1 to suit specific needs.

Commands are entered in the most natural—and efficient—terms. For example, the simple command: "MAKE GRAPH" is all that's needed to take tabular data and produce what's needed on the screen. Also, RS/1 anticipates research needs; it's structured to display data in formats immediately useful to scientists and engineers.

RS/1 runs on Digital's VAX™ and PDP-11™ computers. In fact, Digital has accepted RS/1 in their external software library as a standard package for laboratory data analysis, graphics, statistics and modelling.

RS/1, The Research System. For Relief. For Results.

Bolt Beranek and Newman Inc.

Software in support of scientists and engineers

CIRCLE D216 ON READER CARD

BBN Research Systems—10 Moulton St., Cambridge, MA 02238 **(617) 491-8488**

The Wonderful World Of UNIX

This month our UNIX editor replies to the opinions of Henry Glover two issues ago, outlines a scheme for a distributed network of UNIX systems and discusses how UNIX can be installed as a hidden subsystem on a VMS VAX. UNIX is a trade mark of Bell Laboratories.

MORE OPINIONS?

By Steve Holden, Department of Computer Science, University of Manchester, England

Hello again. One of the frustrating things about writing for an American magazine from England is that the postal delays mean I have to send in my copy at about the same time that the last issue is winging its way across the Atlantic.

Consequently the last column was already flying in the opposite direction when I got to see "An Opinion: UNIX Realities" by Henry Glover in the September issue. At first I was sorry I hadn't been able to include a counterblast last time. Still, second thoughts are proverbially the best and so maybe it was better to wait.

Because those opinions do demand a reply I have had to abbreviate the other material I was promising, but if there's anybody out there who wants to know more please drop the editor a line and I'll see what can be done.

REALITY IS FOR PEOPLE WHO CAN'T HANDLE UNIX?

When I read the Glover article my first reaction was outrage. How dare this person criticize UNIX? What does he know about it anyway? A closer reading revealed that he probably knows at least as much about it as I do, and therefore, his opinions should be respected as based on experience. It does seem to me that an adequate response can be made to each of his points, though, so here goes.

You should, of course, realize that I'm a UNIX enthusiast. I claim not to be a blinkered academic, though, since up to about three years ago I was running a software house (writing BASIC PLUS for RSTS/E). I still like the operating system I was using then, but whatever operating system you decide or are told to use, you cannot afford to be blind to its deficiencies. All operating systems including both RSTS and UNIX have deficiencies. Otherwise what would user groups beat their developers over the head about?

So, what was said about UNIX that I would wish to correct? Let's take each of Glover's points in turn and see what can be said.

1. UNIX is not a good base on which to build an applications environment.

Certainly UNIX is less comprehensive than other operating systems in the sense of not being optimized toward any particular applications environment. This means that you have to build on more layers of software than you might need to do with other systems. The reason that systems houses are switching to UNIX is because this gives them better control over the structure of their systems — there's less to get in the way.

Acquiring a UNIX source license will not stop you having to go to independent software vendors for your applications software. If you're going to develop that yourself then you don't need a source license you need a compiler — and nobody can deny there are plenty of those around even for UNIX!

I don't think UNIX is a one-language system any more than RSTS was, and I see people working on UNIX systems every day programming in PASCAL, C and FORTRAN-77. Since we have never needed any compiler support I can't tell whether we would obtain it grudgingly or not at all. We also have students using cross-assemblers for Motorola and Intel microprocessors and soon hope to be able to compile PASCAL for these and other micros on a PDP-11.

I would argue that UNIX is complete enough to allow such things as indexed-file handlers to be written and integrated into the system. I suppose it depends what you mean by "a good base."

2. With UNIX you always get a UNIXperson.

Personally I think that whenever UNIX is stalled at a new site at least one person BECOMES a UNIXperson. Any company which is prepared to rely on a new employee, especially a college graduate computer scientist, for their sole DP expertise, is asking for trouble.

We're making our terminals even more kind to mankind.

We've done it again.

We've one-upped DEC® with a fully ergonomic version of our CIT-101.

It tilts. It swivels. It has a large 14-inch screen. The detachable keyboard is pure ergonomics right down to its matted key caps.

It's the CIT-101e.

Actually, it makes us more than two-dozen features up on the DEC VT100. Because our CIT-101e has all the extras that DEC asks you buy extra terminals to get.

Features like Advanced Video. A bi-directional printer port. Tutorial soft setup mode.

So you don't have to buy DEC after DEC to get all the features you want. You'll find them all in one CIT-101e.

How's that for kindness?

To learn more, just write or call.

CIE TERMINALS A new company of C. ITOH ELECTRONICS, INC.

New headquarters and Southern California sales office: 2505 McCabe Way, Irvine, CA 92714-6297. (714) 660-1421. Or call toll-free, 1-800-854-5959. In California, 1-800-432-3687. Other regional sales offices: San Jose, CA (408) 247-2393; Cherry Hill, NJ (609) 983-5075; Chicago, IL (312) 992-2346; Houston, TX (713) 777-1640; Atlanta, GA (404) 257-1814; Boston, MA (617) 935-5188; Ontario, Canada (416) 848-1050; Denmark (02) 921100.

CIRCLE D149 ON READER CARD

Wonderful World Of UNIX

The companies which seem to get along best with computers know that to be a commercial success their installation has to be closely controlled. This means a manager who is experienced in DP knows quite a bit about the company's operations, and will not allow random modification of vital company information systems. Of course, in this deal as in all others you get exactly what you pay for, and a new graduate comes cheaper than an experienced professional. You pays your money and you takes your choice.

By the way, I once bought a PDP-11 system from a large (multinational) OEM. It had RP02 lookalike disks, and we used a version of the standalone BACKUP utility, patched by the OEM, to take disk security copies. After fifteen months I discovered that some of my backup copies were garbage.

That problem isn't peculiar to UNIX. Mistakes will always be made. I would also like to take this opportunity to say that I do not consider myself a UNIXperson — I am first and foremost a user, and the reason is that I get my work done quicker on UNIX.

3. No two UNIX systems look alike.

So what? Some RSTS/E systems run with RSX or BASIC2 as the default run-time system. Some BASIC PLUS interpreters have PRINT... USING configured in and others don't. Any respectable software vendor should be able to tell you which operating systems features must be configured in, and if you choose to patch your operating system you must accept that this may lead you down an incompatible path.

UNIXpeople can only do harm when they are given their heads. Hand ten idiots a hammer and one of them might hit you on the thumb with it.

4. UNIX breaks.

If anyone can show me an operating system that doesn't I'd be pleased to give it some publicity. Just the same I don't think that the UNIX disk cacheing is an unreasonable thing.

When I used RSTS I wrote disk routines which did exactly the same sort of thing in BASIC PLUS programs, by cacheing the current data block from each random file to avoid the overhead of multiple disk reads and writes to update each record in a disk block.

Of course, these blocks were not correctly updated if the system crashed. Eventually crashes became troublesome enough to lead us to write repair utilities for affected files. The increase in system throughput allowed us, I estimated, to run an extra three interactive users on a bureau 11/34.

5. UNIX is user-unfriendly.

If we're talking about real users — company presi-

dents, secretaries, clerks and the like — I have yet to meet the operating system that I wouldn't want to bury under a few layers of software. Part of the job of writing computer applications is hiding the bits your users wouldn't understand, and if necessary, you go as far as replacing the login utility so people can say "BEGIN TAYLOR STOCK" rather than "LOGIN 10/20."

UNIX has a concise command style which many people with computer experience find natural and easy to use. Like most people, I have my doubts about the value of using such obscure command names, but with the correct layering this should never be seen by the paying customers.

6. UNIX lookalikes don't.

Caveat emptor! If it doesn't look enough alike, don't buy it. If you don't know how to tell, then you have the right to ask to see everything working together before you buy since it's your money the supplier is trying to get his hands on.

Of course, if that turns out to be impossible then maybe you have too many suppliers involved or the system isn't really the lookalike it claims to be. There will always be overenthusiastic salesmen, and cowboys waiting to take advantage of a gullible public. To them UNIX represents another opportunity, and, whilst I can't condone that sort of thing if it's done deliberately, we do all have a living to make.

It may be that you don't feel these replies are adequate, in which case I'd be pleased to discuss it a bit more. In any case I'd like to thank Henry Glover for spurring me on to defend my favorite operating system.

Hiding UNIX on a VMS VAX.

Last issue I promised that I would give you details of how to sneak up on VMS and hide a UNIX image on its disks. This is a trick which one Alasdair Rawsthorne introduced me to. My schedule and Alasdair's unfortunately haven't coincided, so I can't quote chapter and verse on the exact commands needed.

I hope, though, that dedicated hackers will be able to use the information which follows to achieve the same ends. First a few warnings.

Because this trick depends on being able to modify the VAX bootstrap, as far as I know it will only work on 11/780s — the 750 and the 730 don't have a floppy bootstrap, and so I suppose that their boot images are far less easy to modify. Also, since it means that you have to tweak about with the disk driver tables you have to have a good idea of how and where they are kept and what they look like. You probably need a source license.

In short, this is probably for wizards and gurus only, but it's worth putting about because I can see no logical reason why the same techniques shouldn't be used to mix any operating systems. The usual warnings about not taking any responsibility for screwups apply — I don't want people complaining that I didn't warn them!

The technique first of all demands that you create a smallish file on the VMS disk. It should be a contiguous file occupying a whole number of cylinders; if it isn't flagged as

Wonderful World Of UNIX

non-movable you will run the risk of destroying much of the contents of the disk pack. This file will become the UNIX virtual disk later on, although to VMS it will look just like an ordinary file.

Second, you write a modified (gasp eek!) UNIX boot image on to the bootstrap area of this virtual disk. This boot image should contain a disk driver whose tables have been patched to point to the VMS file containing the UNIX virtual disk. When this image is loaded under bootstrap control it believes that it only has access to the single file which you created at the start of the process.

UNIX's disk routines are quite happy to treat this as a complete, albeit rather small, disk pack. You thus have a single file under VMS which becomes a whole UNIX disk.

The last stage is to produce a modified copy of the bootstrap floppy which boots its initial core image from the virtual disk instead of the usual bootstrap area. When the system boots using this new floppy it then reads in the modified UNIX image from the bootstrap area of the virtual disk, and UNIX comes up. No disk changes, no chance of messing up the VMS filestore, and no need to tell your VMS manager that you're coming in and running UNIX in the middle of the night!

Another colleague has produced software which allows VMS to access the UNIX virtual disk to pull files off into the VMS world or do inter-UNIX tape transfers even when VMS is running.

Naturally I don't suggest that this is a suitable method for a commercial system, but it's very convenient to have the two systems living on the same disk. It seems to be a little like The Bridge, which I understand allows CP/M to co-exist with PDP-11 operating systems. Since both VMS and UNIX (in our case Berkeley 4.1) run on the same processor, however, there is no need to add a special board containing the other processor. Unlike in The Bridge the two systems cannot run concurrently.

Still, you can't have everything. Alasdair's still trying to figure out whether it's worth writing a proper description of the technique, so let me know if you want the real lowdown. A final disclaimer — I told you earlier I wasn't a true UNIX-person, and so any errors or omissions in the above section are mine and not his. Unlike me, Alasdair knows enough about the guts of UNIX to get away with this kind of thing.

UNIX United

Finally for this month, a short word about this interesting extension to the UNIX filing system. Most people are aware that UNIX uses a tree-structured directory format. The UNIX United modification allows a set of UNIX systems which are connected by some sort of communications network to share and freely access each other's file stores.

As far as I know this idea originated for Version 7 at Newcastle University in England. It is now being followed up by several other UNIX researchers, notably Roy Campbell at Urbana Champaign, who is busy incorporating the same thing into the Berkeley system. You may also see it referred to as "United UNIX" or "The Newcastle Connection."

To give you an idea of how it looks to the user (please don't ask me how it works internally, because I'd only be guessing if I tried to tell you), first let me describe how the tree-structuring looks on a standard UNIX filestore. Suppose I am identified as **steve** to a UNIX system. The **login** utility program refers to a file which contains the name of all users, and for each user the name of a directory file (amongst other things).

Typically my directory would be **/user/steve**, although this is less likely to be true on systems more recent than Version 7. So **login** makes my current directory **/user/steve** as it logs me in.

I can now use the special name ". ." to refer to the directory which contains my directory. So, for example, I could get a listing of Dave's files with the command

l s ./dave

because the directory above /usr/steve is /usr, so the above filename is equivalent to /usr/dave. It's reasonably simple, I hope (otherwise how come I understand it?).

The little bit of magic which UNIX United works is to give each machine in the network a name, and to put a conceptual or virtual directory above the whole network. So, supposing I were logged in on VAXA and my current directory were /usr/steve, I could get a listing of Dave's files on VAXB by using the command

l s ../../../VAXB/usr/dave

which looks a bit confusing but actually isn't that difficult. The key thing to remember is that each ". ." takes you up a level in the directory tree.

So ". ." is /usr on VAXA. ". ./. ." is the root directory on VAXA. This makes ". ./. ./. ." the UNIX United virtual directory, and from this point on the directory specification is straightforward. I think the whole scheme is pretty neat, although I couldn't recommend it for commercial sites at present.

Still, if research has any useful function for the commercial computing world it is to point the way forward. We are probably still some time away from seeing this kind of system properly supported on a local area network like Ethernet, but I can hardly wait.

To harp back to the earlier opinions, one of the nicest things about UNIX is that its elegance and simplicity encourage research directions like this. And if anyone wants to prove me wrong by telling me that UNIX United is already or soon to be commercially available I'd be pleased to give him a small amount of free publicity in this column.

The Competition

Maybe UNIX users really are read-only. So far I haven't had any urgent messages from the editorial offices asking me to verify the answer to last issue's competition. Consequently I shan't be setting up another one yet.

I just had a really ugly thought. Maybe I didn't manage to get the note on to USENET properly, and thousands of Stateside hackers have been wasting their time looking for it. Perhaps I'll ring a friend in the US just to make sure!

NOW ON DEC PROFESSIONAL RAINBOW IBM PC

PDP-11, VAX, 68000, 8088

Test your word processing I.Q.

And discover how the advanced features of LEX-11™ can increase your w-p capabilities.

THE CHALLENGE:

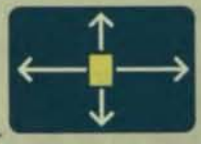
Almost any word processing software package will have such features as menus, editing, spelling error detection, list processing, cut and paste, automatic word wrap and automatic pagination. . .

. . .But can you tell the difference between run-of-the-mill software and a system that is really special?

EEC SYSTEMS offers you this challenge! Test your word processing I.Q.

THE QUESTIONS:

1 EDITING



Which kind of editing operation is quickest to execute and easiest on the eyes of the word processing user?

a) Full screen editing allowing for easy cursor movement around the screen?

b) Moving the cursor around by doing a line count?

c) Editing on the bottom line of text only?

2 DOCUMENT LAYOUT

Whatever document format you choose. . .you want to see what the finished article will look like. Should you. . .

a) View it on the screen as it would come out of the printer?

b) Run it through a pre-processor to see what it looks like and then if you like it, print it?

3 KEYSTROKES

Using a well designed w-p system, how many keystrokes should it take to execute the most often used w-p functions?

a) One easy stroke with no codes?

b) Two or more with complex w-p codes?

c) Three or more?

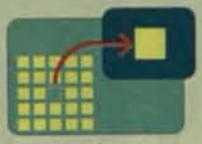
4 FLEXIBILITY

As the business manager of your company, you would like to find w-p software that you can tailor to your company's specific needs. Should you. . .

a) Look for w-p software that allows you to change and add menus, and change function keys?

b) Write your own custom software?

5 RETRIEVAL



If you want to retrieve information quickly from a large database, which w-p software should you choose?

a) One that can access a particular record by going to it directly?

b) One that searches through all the records on the database sequentially until it finds the right one?

6 COMPATIBILITY

As a manager of MIS, you want a w-p system that can be integrated with other application software. Should you choose w-p software with. . .

a) ASCII formatted files?

b) Software which requires non-printing characters in it's file system?

7 MATH

Your company has a number of financial applications and is looking for a w-p package with math capabilities. Should you choose. . .

a) On screen calculating allowing for editing, storing and recall of equations, calculations integrated with your word processing applications?

b) Software where the math capabilities are tied to the list processing module?

c) A separate math package?

THE ANSWERS:

If you answered "a" to all of the questions above, go to the top of the class. Chances are that you already use LEX-11 or are about to buy it. You know that LEX-11 is the ONLY word processing software that combines these advanced features. . .and more. . . in one software package.

PUT OUR SOFTWARE TO THE TEST

Call for a demonstration by modem in your office today.

(617) 443-5106 **(617) 443-6376**

EEC SYSTEMS INC.
Millbrook Park
327/E Boston Post Rd
Sudbury, MA 01776

LEX-11, LEX-68 and LEX-88 are available under the following operating systems: CP/M,■ MS-DOS,▫ RT-11,* TSX-Plus,* RSX-11M,* P/OS,* RSTS/E,* IAS,* UNIX† (68000,□ 8088†† PDP-11* and VAX*), IDRIS,** VMS* . . .and it works in exactly the same way, with the same features for all operating systems, with your existing peripheral equipment.

Wonderful World Of UNIX

EOT

So, as Holden slowly sinks in the West another UNIX column comes to an end. If I wanted to get paranoid I could believe that there's nobody out there reading all this stuff.

Hopefully the next couple of months will see some letters coming in for me to respond to. I am especially keen to publish any neat tricks with the shell command language.

If no mail arrives, I suppose I'll just have to become more and more outrageous until someone is forced to write in just to complain that THE DEC* PROFESSIONAL is lending its columns to a basket case. Anyway, until the next issue, try to stay away from batch systems.

Editor's Note: Many of our readers were stirred by Henry Glover's "An Opinion: UNIX Realities," published in September, 83.

One in particular, Arthur Metz, was moved to reply rather eloquently to Mr. Glover's charges, and deserves equal time.

A REBUTTAL — UNIX REALITIES

By Arthur Metz, Sopher Computer Consultants, Los Angeles, CA

Henry Glover's article, "An Opinion: Unix Realities," published in the September issue demands a rebuttal. I am not a UNIXPerson (sic); I did not use UNIX at school (UCLA, CalTech); in my professional career, I have used everything from MVS to VMS, MPX to MPE to MP/M. And yes, UNIX. Several of Glover's criticisms are valid, but others are totally misdirected. Let's examine his "realities."

"Reality Number 1: UNIX is not a good base on which to build an applications environment." Well, maybe. Depends on the application. More important, UNIX probably is a good base on which to build it, but may be a poor base on which to run it. (No time-sharing system should be used for real-time applications, like air traffic control or Pac-Man.) But let's look more closely at the observations which lead Glover to this reality.

1a) "UNIX is oriented to text processing applications." True. That's what makes it good for project development, documentation, etc. (Yet, paradoxically, UNIX does not have a decent standard screen editor.)

1b) "The file system has no record access mechanism." Ahem. First, what I consider UNIX's file system is its directory hierarchies, and they are super. As for record access, this is true, but fairly irrelevent. What Glover calls "basic record sizes" of 1 or 512 bytes is in fact a totally transparent blocking scheme. Because it is transparent, you can ignore it and use any logical record size you want. You can read n bytes, to an EOL character, or whatever. You can also do random positioning to any byte in the file. I submit that this flexibility is greater than generally available elsewhere.

1c) ". . . indexed files, variable length records, and indexed sequential access are all foreign to UNIX." Yes, no, yes, in that order. Glover seems to want a manufacturer to supply its own versions of all three access methods. May I suggest IBM? And then, may I point to the plethora of second-source suppliers who try to fix IBM's defects? The strength of UNIX is that it's easier to add these features to a stripped system than to overcome a feature-laden, low-performance environment.

1d) "UNIX . . . [is] aimed at the C language program developer." True. C and UNIX are a great environment to write systems software. They are lousy at accounting packages (no BCD data type, inadequate data security), real-time process control (too much swapping), or manipulating large matrices (no built-in floating point hardware). Don't criticize a fork because you can't drink soup with it.

1e) "Layered products...must be acquired from a second source." True. It's called unbundled software. "All this second-sourcing becomes a problem in support." Maybe. But this is hardly unique to UNIX. If all your VAX software comes from DEC, then you're ignoring good data base, word processing and spreadsheet software.

"Reality Number 2: with UNIX you always get a UNIXperson." Not in my experience. Here, Glover's complaint seems to be that the particular UNIXperson was incompetent. Well, that happens. But I find his story of the bad backups hard to believe. In three years it was NEVER necessary to read a backup tape (or disk)? That speaks pretty well of UNIX. And backups are often done in a compressed format (e.g., tar). Presumably that's why Mr. Glover found "total garbage."

If your site installs a program without testing it, lets it run for three years unchecked and then finds the output

. . . continued on page 69

COMMITTED TO DEC?

So are we. And, we're committed to the individual systems buyer too. We give the little guy the edge he just can't get anywhere else. We understand the system builder's time constraints, and we're flexible enough to work with them.

We bring the latest technology to our added value DEC systems long before anyone else. And, we relieve you of the complex, time-consuming task of searching for and evaluating new high-performance products and system possibilities.

Plus, we give you considerably faster turnaround. With the Cambridge Digital "Edge" you get a fully integrated, PDP or VAX system in less than 10 days. And, your system will be up and running on delivery with a full complement of fully supported software and peripherals. The best, most advanced products on the market today. All tested and ready to go.

So, whether you want a fully integrated pre-packaged system or you want to mix and match system components, Cambridge Digital can give you the performance you need in an economical package, ready to go the day you get it. That's what the Cambridge Digital "Edge," is all about.

To receive our DEC PDP-11 based system catalog including a description of the seven guarantees you get when you get The Edge, call or write.

Main Office Dept. 7504
P.O. Box 568, 65 Bent St.
Cambridge, Massachusetts 02139
Telex 92-1401/COMPUMART CAM
800-343-5504. In Mass. call 617-491-2700
New York District Office 516-935-3111

Cambridge Digital
DIVISION OF COMPUMART

The Edge in System Integration.

800-343-5504
In Massachusetts call 617-491-2700

CIRCLE D144 ON READER CARD

DIGITAL INTRODUCES ADVANCED VT200 FAMILY OF TERMINALS

BOSTON, MA, November 15, 1983 — Digital Equipment Corporation today introduced a sleek new line of video display terminals with lower price tags and more functions than its best-selling VT100 terminals. The new VT200 video terminal family boasts three ergonomically designed models with features ranging from advanced text handling to both text and color graphics capabilities.

The VT200 family consists of the VT220, VT240 and VT241 models priced at $1,295, $2,195, and $3,195, respectively. Customers can order directly from the company or through its distribution channels, such as Authorized Terminal Distributors and Industrial Distributors. Initial deliveries are immediate.

Arthur T. Campbell, Terminals Product Group Manager, said the advanced features and aggressive pricing of the new terminal series will enable the company to increase its dominance in the terminals market. "Our customers and distributors have already indicated to us they believe the demand for the VT200 terminals will even surpass that of VT100s. Therefore, we are planning production capacity so that we can ship at least as many VT200s as VT100s during the balance of this fiscal year," he said.

The company said it will continue production of VT100 terminals for at least two years, and longer if demand continues. The VT100, introduced in 1978, has become the industry's best selling ASCII display terminal and spawned a huge emulation market. Digital has shipped more than a half million VT100 units.

Campbell said VT200 terminals have full VT100 emulation capabilities. This enables customers to add VT200s economically to systems currently supporting VT100 terminals without changing software, he said.

DIGITAL INTRODUCES...

The new terminals are targeted for a variety of applications including data entry, office automation, electronic mail, interactive computing, communications, networking and business graphics. Markets include Original Equipment Manufacturers (OEMs), systems integrators, and end-users in business, engineering, medicine, laboratories, and education.

The Family Members:

• The VT220 terminal is a two-piece monochromatic text unit consisting of an ergonomically styled keyboard and 12-inch, non-glare monitor. The compact, wedge-shaped video monitor is available with white, green or amber screens and contains all logic circuitry.

• The VT240 terminal has all the text features of the VT220 plus bit-map graphics capabilities for business graphics applications. The monochromatic terminal supports two high-level graphics instruction sets: Digital's ReGIS (Remote Graphics Instruction Set) and Tektronix' 4010/4014 graphic protocols.

The terminal is a three-piece assembly with monitor, keyboard and system box containing the terminal's logic. Digital has available application software, such as the DECgraph and DECslide packages, to support the graphics features of the

V3.1

THE DEC* PROFESSIONAL

THE MAGAZINE FOR DEC USERS

Please print all information requested. Please answer all questions.

Area Code ____ Office Telephone ____

Do you wish to receive (continue to receive) a **FREE** subscription to The DEC Professional

☐ Yes ☐ No Date X _____

☐ Check here if renewal.

☐ Canada $25 (US$) ☐ Foreign $50 (US$) check must accompany this card.

X _____

Signature required to qualify does not obligate me or my company.

MAKE ANY NECESSARY ADDRESS CHANGES HERE

NAME _____

TITLE _____ MAIL STOP _____

COMPANY _____

ADDRESS _____

CITY/STATE/ZIP _____

1. Your own primary area EDP activity (check **ONE** only)
- ☐ **User** of Computing or Data Processing Equipment
- ☐ **Manufacturer** of Computing or Data Processing Equipment

2. COMPANY MANAGEMENT
- ☐ A. **General & Corporate:** President/Owner/Director/Chairman/Partner, Vice President/General Manager.
- ☐ B. **Financial:** Comptroller/Treasurer, Vice President Finance, Chief Accountant.
- ☐ C. **Administrative Services:** Vice President Operations, Director or Manager Purchasing/Personnel/Office, Directors or Managers of Microfilm.
- ☐ D. **Marketing:** Vice President Sales/Sales Manager, Director or Manager Marketing.

COMPUTER/SYSTEMS OPERATIONS
- ☐ E. **Management:** Vice President EDP, Director/Manager/Head/Chief of EDP/MIS, Director or Manager of Programming, Director or Manager Systems & Procedures, Director or Manager Communications, Director or Manager EDP Operations.
- ☐ F. **Supervisory:** Supervisor Equipment/Analysis, Chief/Lead/Senior Programmer, Chief/Lead/Senior Mathematician, Senior Systems Analyst.
- ☐ G. **Staff:** Analyst/Systems Analyst, Mathematician, Programmer, Machine or Computer Operator, Software Designer, EDP Auditor.

ENGINEERING/MANUFACTURING
- ☐ H. **Management:** Vice President or Director of R&D, Vice President or Director Engineering, Plant or Production Manager/Director, Chief Engineer.
- ☐ I. **Staff:** Member of Technical Staff/Group Leader, Systems Engineer, Design/Production/Research Engineer, Maintenance Engineer, Application Engineer, Other Engineer.

- ☐ J. **CONSULTANT**
- ☐ K. **EDUCATOR:** Educator (instruct EDP) and Educator (use EDP)
- ☐ L. **OTHER** _____

3. Primary end product or service of your firm at this location. (Insert code)
- ☐ A. Manufacturing: Computers, Data Processing and Peripheral Equipment.
- ☐ B. Other Manufacturing and Processing.
- ☐ C. Government: Federal, State, Municipal, including Military.
- ☐ D. Educational: Colleges, Universities, and other Educational Institutions.
- ☐ E. Computer and Data Processing Services including Software Services, Service Bureaus, Time Sharing and EDP Consultants.
- ☐ F. Finance: Banking, Insurance, Real Estate, Securities, and Credit.
- ☐ G. Trade: Wholesale and Retail.
- ☐ H. Medical and Legal Services.
- ☐ I. Transportation Services: Land, Sea, and Air.
- ☐ J. Utilities: Communications, Electric, Gas, and Sanitary Services.
- ☐ K. Construction, Mining and Agriculture.
- ☐ L. Other Business Services, such as Business Managment Consultants, Hotels, Amusements and Non-Profit Organizations

4. Total number of employees in the entire company/institution/governmental department.
- ☐ 1. 10,000 or more
- ☐ 2. 5,000 to 9,999
- ☐ 3. 2,500 to 4,999
- ☐ 4. 1,000 to 2,499
- ☐ 5. 500 to 999
- ☐ 6. 250 to 499
- ☐ 7. 100 to 249
- ☐ 8. 6 to 99
- ☐ 9. Fewer than 6 (Check one only)

5. CPU (enter any you have)

VAX 782	☐ 1	PDP 24	☐ 25	LSI-11 Falcon	☐ 46
780	☐ 2	23+	☐ 27	T-11	☐ 48
750	☐ 10	23	☐ 29	PDP8	☐ 61
730	☐ 12	35/40/45/60, etc.	☐ 30	WPS	☐ 62
MICROVAX	☐ 14			VT180	☐ 63
PDP 70	☐ 31	MICRO-11	☐ 33	10/20	☐ 64
44	☐ 32	LSI-11 03	☐ 42	PDP(other)	☐ 65
34	☐ 34	11/73	☐ 44		

PROFESSIONAL		
350	☐ 13	
325	☐ 15	
RAINBOW		
100	☐ 83	
100+	☐ 84	
DECMATE		
I	☐ 85	
II	☐ 86	

6. DISK TYPES (check any)

RX	☐ 101	DECNET	☐ 109
RL	☐ 102	ETHERNET	☐ 124
RC	☐ 106	STREAMERS	☐ 125
RK	☐ 107	TU/TE TAPES	☐ 126
RP	☐ 108	DBMS	☐ 127
RM	☐ 111		
RA	☐ 113		
RF/RS	☐ 115		
other	☐ 119		

7. DISC SUPPLIER (check either or both)

DEC	☐ 120	HSC50	☐ 122
NON/DEC	☐ 121		
I also have			

8. TOTAL DISK STORAGE (check one)

< 5MB	☐ 1	101-200MB	☐ 4
5-10MB	☐ 2	201-500MB	☐ 5
11-100MB	☐ 3	501-1GB	☐ 6
		>1GB	☐ 7
		>2GB	☐ 8

9. OP/SYS (check any you use)

VAX/VMS	☐ 1	MS/DOS	☐ 13
RSTS/E/CTS500	☐ 2	CONCUR CP/M	☐ 14
RSX 11	☐ 3		
RT11/CTS300	☐ 4		
MUMPS	☐ 5		
P/OS	☐ 6		
CP/M	☐ 7		
TSX	☐ 8		
UNIX	☐ 9		
TOPS	☐ 10		
IAS	☐ 11		

10. I am an OEM ☐ YES ☐ NO

READER INQUIRY CARD — JANUARY 1984 (Card expires March 1984)

Circle numbers of advertisements, new literature or new products here:

D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16	D17	D18
D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D29	D30	D31	D32	D33	D34	D35	D36
D37	D38	D39	D40	D41	D42	D43	D44	D45	D46	D47	D48	D49	D50	D51	D52	D53	D54
D55	D56	D57	D58	D59	D60	D61	D62	D63	D64	D65	D66	D67	D68	D69	D70	D71	D72
D73	D74	D75	D76	D77	D78	D79	D80	D81	D82	D83	D84	D85	D86	D87	D88	D89	D90
D91	D92	D93	D94	D95	D96	D97	D98	D99	D100	D101	D102	D103	D104	D105	D106	D107	D108
D109	D110	D111	D112	D113	D114	D115	D116	D117	D118	D119	D120	D121	D122	D123	D124	D125	D126
D127	D128	D129	D130	D131	D132	D133	D134	D135	D136	D137	D138	D139	D140	D141	D142	D143	D144
D145	D146	D147	D148	D149	D150	D151	D152	D153	D154	D155	D156	D157	D158	D159	D160	D161	D162
D163	D164	D165	D166	D167	D168	D169	D170	D171	D172	D173	D174	D175	D176	D177	D178	D179	D180
D181	D182	D183	D184	D185	D186	D187	D188	D189	D190	D191	D192	D193	D194	D195	D196	D197	D198
D199	D200	D201	D202	D203	D204	D205	D206	D207	D208	D209	D210	D211	D212	D213	D214	D215	D216
D217	D218	D219	D220	D221	D222	D223	D224	D225	D226	D227	D228	D229	D230	D231	D232	D233	D234
D235	D236	D237	D238	D239	D240	D241	D242	D243	D244	D245	D246	D247	D248	D249	D250	D251	D252
D253	D254	D255	D256	D257	D258	D259	D260	D261	D262	D263	D264	D265	D266	D267	D268	D269	D270
D271	D272	D273	D274	D275	D276	D277	D278	D279	D280	D281	D282	D283	D284	D285	D286	D287	D288
D289	D290	D291	D292	D293	D294	D295	D296	D297	D298	D299	D300	D301	D302	D303	D304	D305	D306
D307	D308	D309	D310	D311	D312	D313	D314	D315	D316	D317	D318	D319	D320	D321	D322	D323	D324
D325	D326	D327	D328	D329	D330	D331	D332	D333	D334	D335	D336	D337	D338	D339	D340	D341	D342

FOR PROMPT ACTION ON ITEMS CIRCLED, FILL IN SECTION BELOW: (Please use label from magazine if available)

NAME _____ TITLE _____

COMPANY _____

ADDRESS _____ SUITE _____

CITY _____ STATE _____ ZIP _____

TELEPHONE ( ) _____

☐ I wish to subscribe to THE DEC* PROFESSIONAL, please send information.

V3.1

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 16 AMBLER, PA

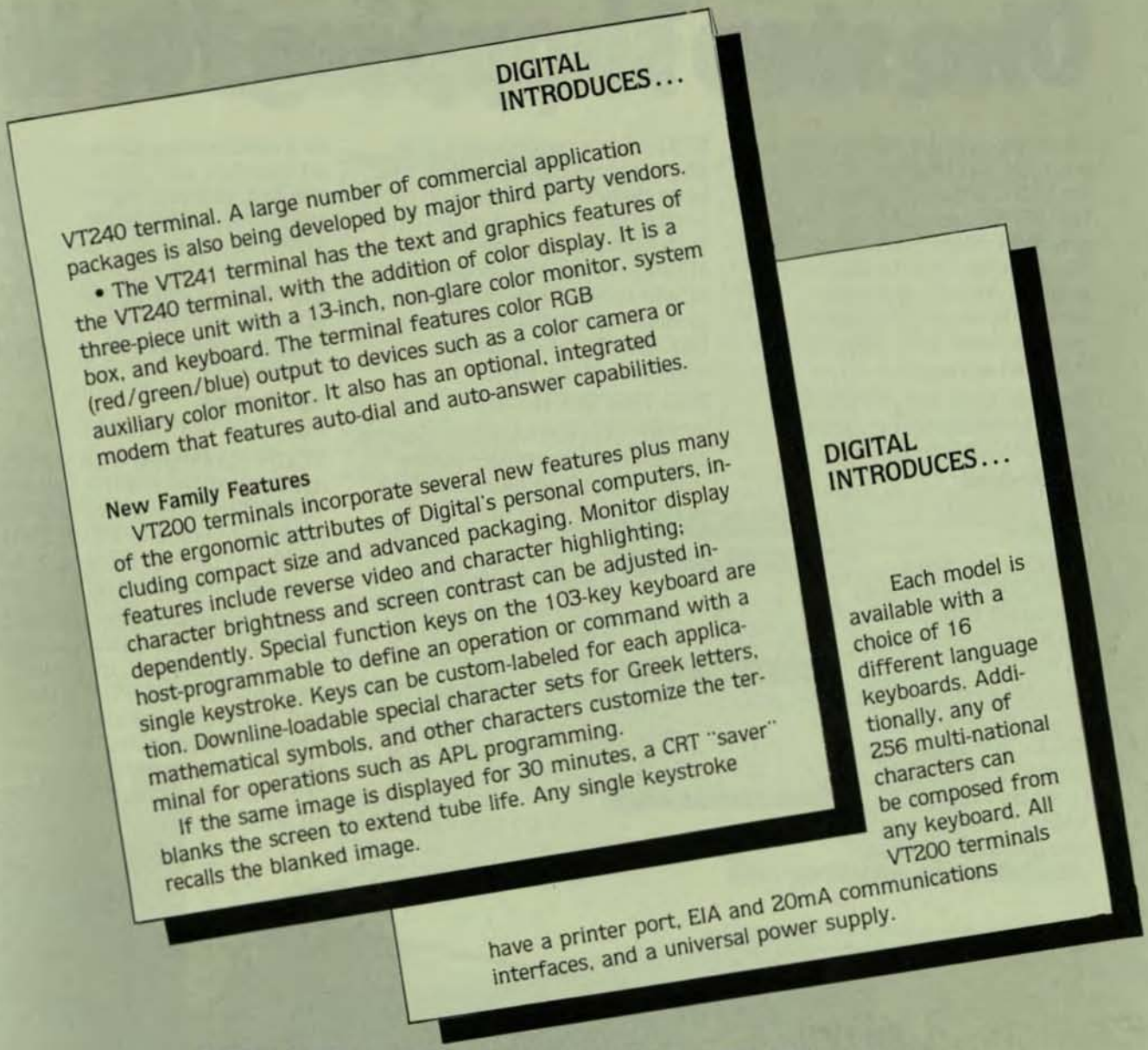
THE DEC PROFESSIONAL
P.O. BOX 362
AMBLER, PA 19002-0362

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 16 AMBLER, PA

POSTAGE WILL BE PAID BY ADDRESSEE

THE DEC PROFESSIONAL
READER INQUIRY
P.O. BOX 362
AMBLER, PA 19002-0362



THE NEW DEC VT200 TERMINAL SERIES

By Allen A. Watson, The Record, Hackensack, NJ

In mid-November, 1983, Digital Equipment Corporation officially announced the new VT200 series terminals that they had previewed earlier in the Boston DECTown trade show and again at the DECUS Symposium in Las Vegas during October.

The best news about this new terminal series is the list prices. They are:

VT220	$1080
VT240	$1980
VT241	$2980

The keyboard is priced separately because it comes in several foreign language flavors. If you add the keyboard price, $215, to the above terminal prices you arrive at a final list price per unit of:

VT220 with keyboard	$1295
VT240 with keyboard	$2195
VT241 with keyboard	$3195

These new terminals are classy looking. If you have seen a DEC Personal Computer you know what the VT200 series looks like. The keyboard and monitor are identical to those used with the PCs. The same low profile keyboard, with the same key arrangement, the same tapered, modernistic monitor. The terminals are all smaller and take less desk space than a VT100 — yet all have at least a 12-inch screen. With the VT100, DEC set a standard for terminals. They aim to do so again with the VT200s and in my opinion, they will succeed.

The VT200s start where the VT100 left off. They include all of the familiar VT100 features such as pan scroll, 132-character or 80-character lines, numeric keypad and special function keys. They are fully compatible with the VT100. Presumably, you can simply plug them into your system and run. And as for reliability, Digital is asking only $9 per month for on-site maintenance. Obviously they don't expect the terminals to have many problems.

One stop shopping for VA

No other supplier offers such a winning combination of products for VAX computers. Reliable? You bet! MTBF figures for our VAX products range from 40,000 to 85,250 hours. Add to that competitive pricing. And we're second to none in technical support, with third party service available as required.

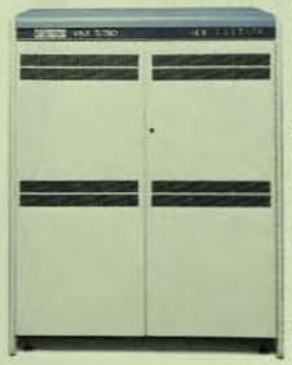
For disk, tape and communications controllers, the Emulex lineup looks like this:

Disk Products.

FOR THE VAX UNIBUS...

SC12/V—Emulates DEC's RK711 controller combined with multiple RK07 drives on the VAX-11 Unibus.

SC21/V—Emulates DEC RM03 (80 MByte) and RM05 (300 MByte) storage subsystems. Includes Emulex VMS/VM software driver/diagnostic package.



SC31—A low cost solution that allows you to install and operate large capacity disk drives on the Unibus of any VAX. Handles drives with high transfer rates of 1.8 MBytes per second in the 500 MByte range. Gives the same or greater storage capability than DEC Massbus installations at a fraction of the cost.

FOR THE VAX-11/750...

SC750—This software-transparent, single-board controller allows you to add up to four large disk storage units (80 to 675 MBytes) directly to the internal CMI bus. The SC758 lets you add up to eight drives of storage off a single controller.

FOR THE VAX-11/780...

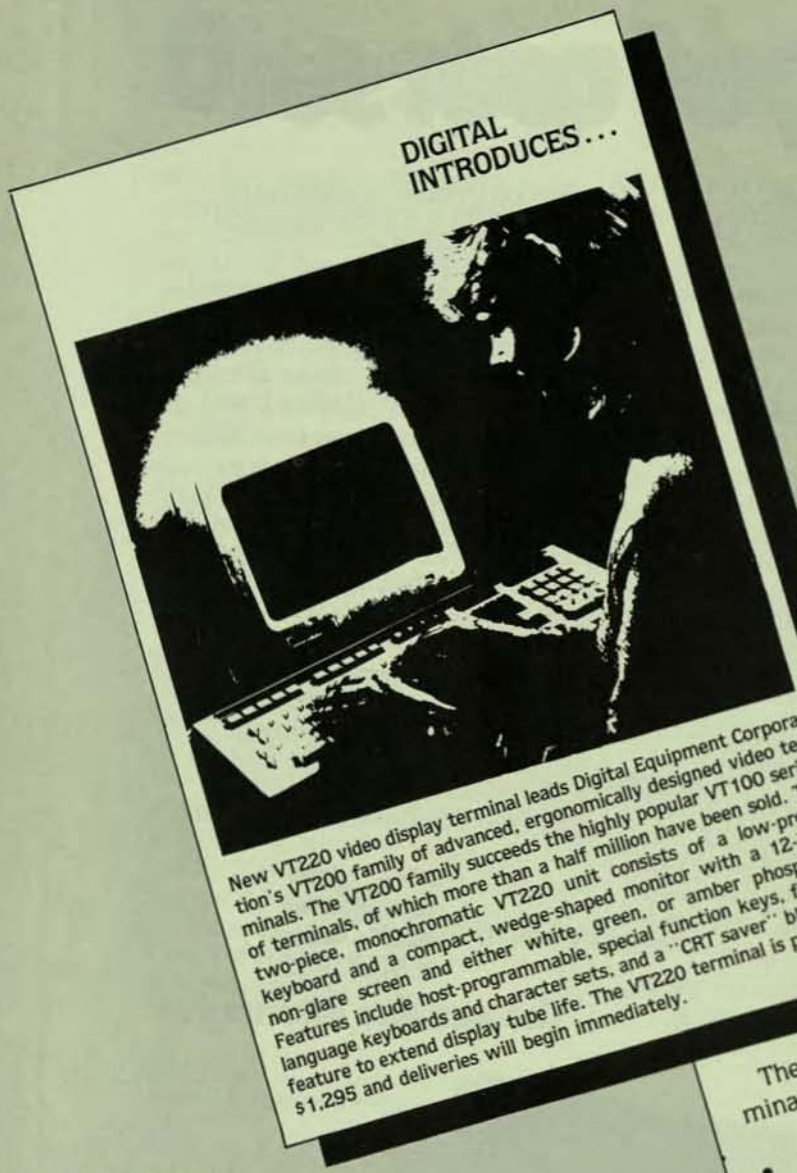
V-Master/780—A mass storage adapter that houses one or two SC780 disk controllers, TC7000 tape controllers or a combination thereof. Provides an interface and control through the Synchronous Bus Interface (SBI) of your VAX-11/780. Each SC780 disk controller supports up to four disk drives (80 to 675 MBytes). The SC788 is also available to fit in the V-Master/780 chassis and supports up to eight disk drives.

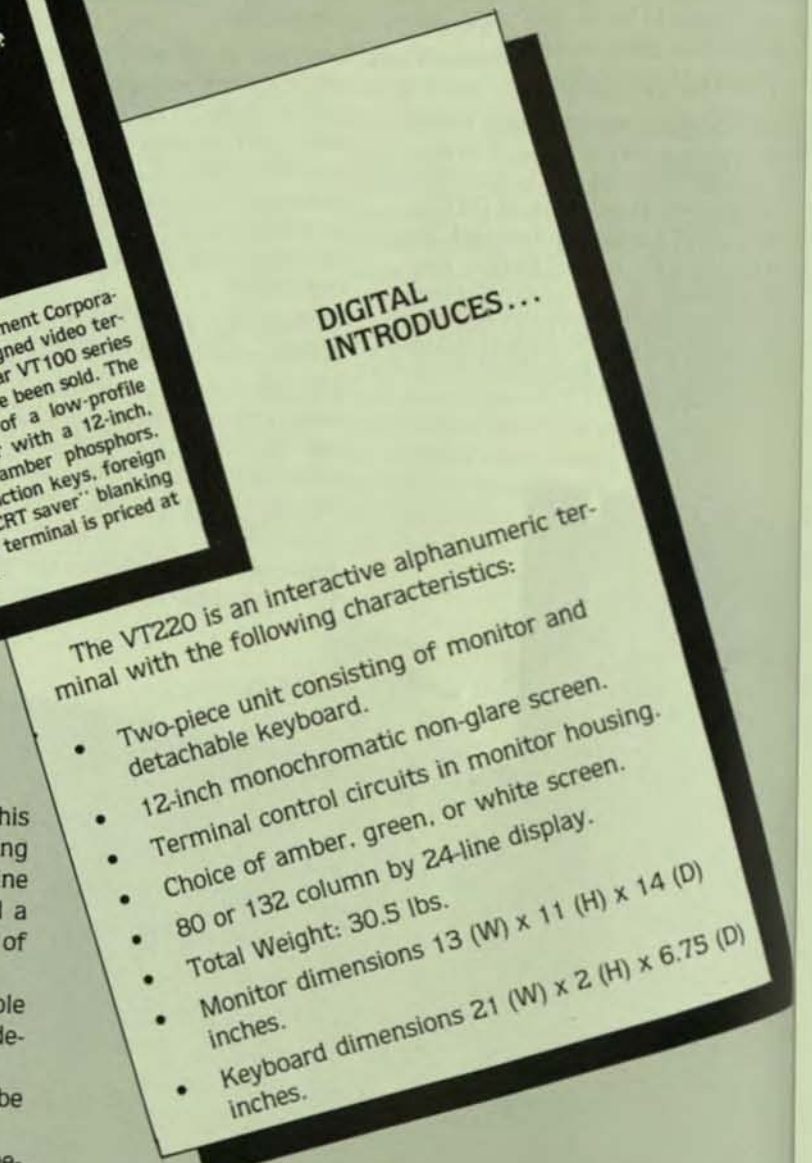
Tape Products.

FOR THE VAX UNIBUS...

TC11/V—Combines with any standard tape drive and the Emulex VMS/UT software driver/diagnostic package to emulate DEC's TM11/TU10 and provide reliable, economical tape storage on all VAX-11s.

TC12/V—Handles every industry-standard "Pertec" formatted half-inch tape transport, including conventional NRZ/PE start/stop and 1600/3200 bpi start/stop streaming tape drives. Provides software transparent emulation of DEC's TS11 subsystem on all VAX-11s.





THE VT220

The bottom line of the VT200 series is the VT220. This terminal has all the features of a VT102 terminal, including Advanced Video Option (bold, blink, reverse and underline modes, with a full twenty-four 132-character lines) and a printer port, with limited graphics capability in the form of a graphics character set.

In addition, the VT220 offers fifteen programmable function keys that can be utilized for special, application-dependent operations.

The character set consists of 256 characters that can be downline-loaded from the host computer.

A selective erase feature allows you to erase only specific fields on the screen; you can blank portions of a line instead of just the entire line.

One option that drew applause when it was announced was an English Set-up mode, as opposed to the four to six sets of four binary zeros and ones that are used on the VT100, forcing the user to read a reference card every time he wished to change a mode setting. Selection of light or dark background, block or line cursor, autorepeat, and jump or smooth scrolling are among the features that are now controlled by responding to plain English prompts on the screen.

The VT220 doesn't understand just English. One reason the keyboard is priced separately is that many languages (French, Spanish and German are among them) all have their own keyboard and special character set, with related Set-up prompts.

The VT220 monitor is equipped with a "CRT saver" that blanks the screen if the terminal is inactive for more than thirty minutes. This may seem a small thing, but when we installed a similar feature 110 non-DEC terminals in our newsroom, it proved so effective that life expectancy of the CRTs tripled. That is a non-trivial savings in real dollars.

How to get Supercomputer Performance with your VAX.

You can get supercomputer performance for structural analysis applications for a fraction of the cost of a supercomputer or a large mainframe.*

The FPS-164 from Floating Point Systems, coupled with your VAX, can give you the high-speed computing you need for scientific and engineering applications.

The FPS-164, designed with 64-bit precision and up to 7.25 Megawords (58 Megabytes) of memory, is especially suited for finite element modeling.

For example, coupling the FPS-164 with your VAX will allow you to run ANSYS up to 17 times faster than before. Over ten software packages for structural applications are available to run on the FPS-164 including MSC/NASTRAN, ABAQUS, ADLPIPE and SDRC SUPERTAB.

For more information on what's available with the FPS-164, write us.

FLOATING POINT SYSTEMS, INC.

P.O. Box 23489
Portland, OR 97223
(503) 641-3151
TLX: 360470 FLOATPOIN BEAV

To: Marketing Communications, S586
Floating Point Systems, Inc.
P.O. Box 23489
Portland, Oregon 97223

I'd like to know more about the FPS-164.

Name _____

Position _____

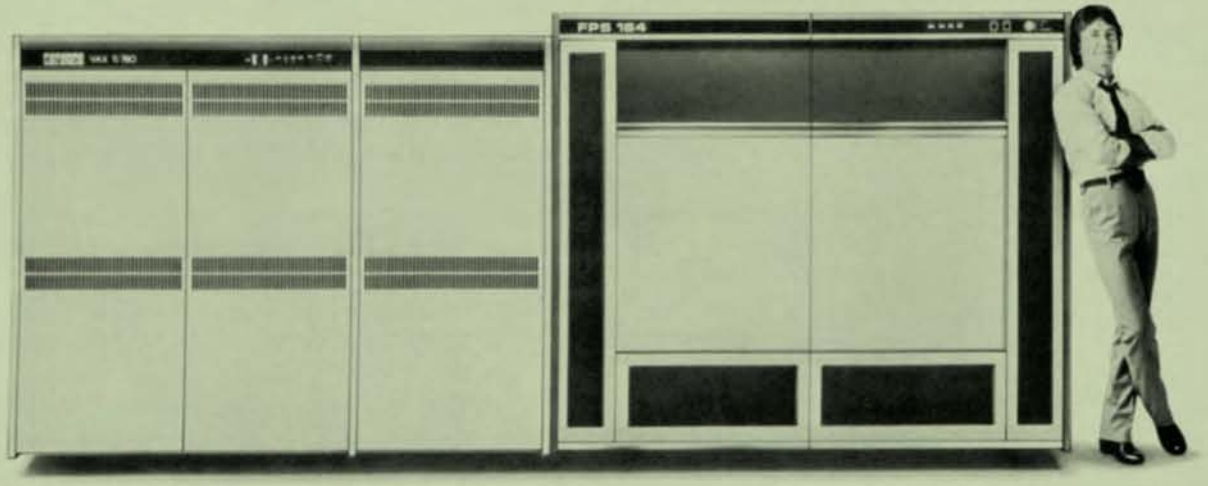
Company _____

Address _____

_____ Phone _____

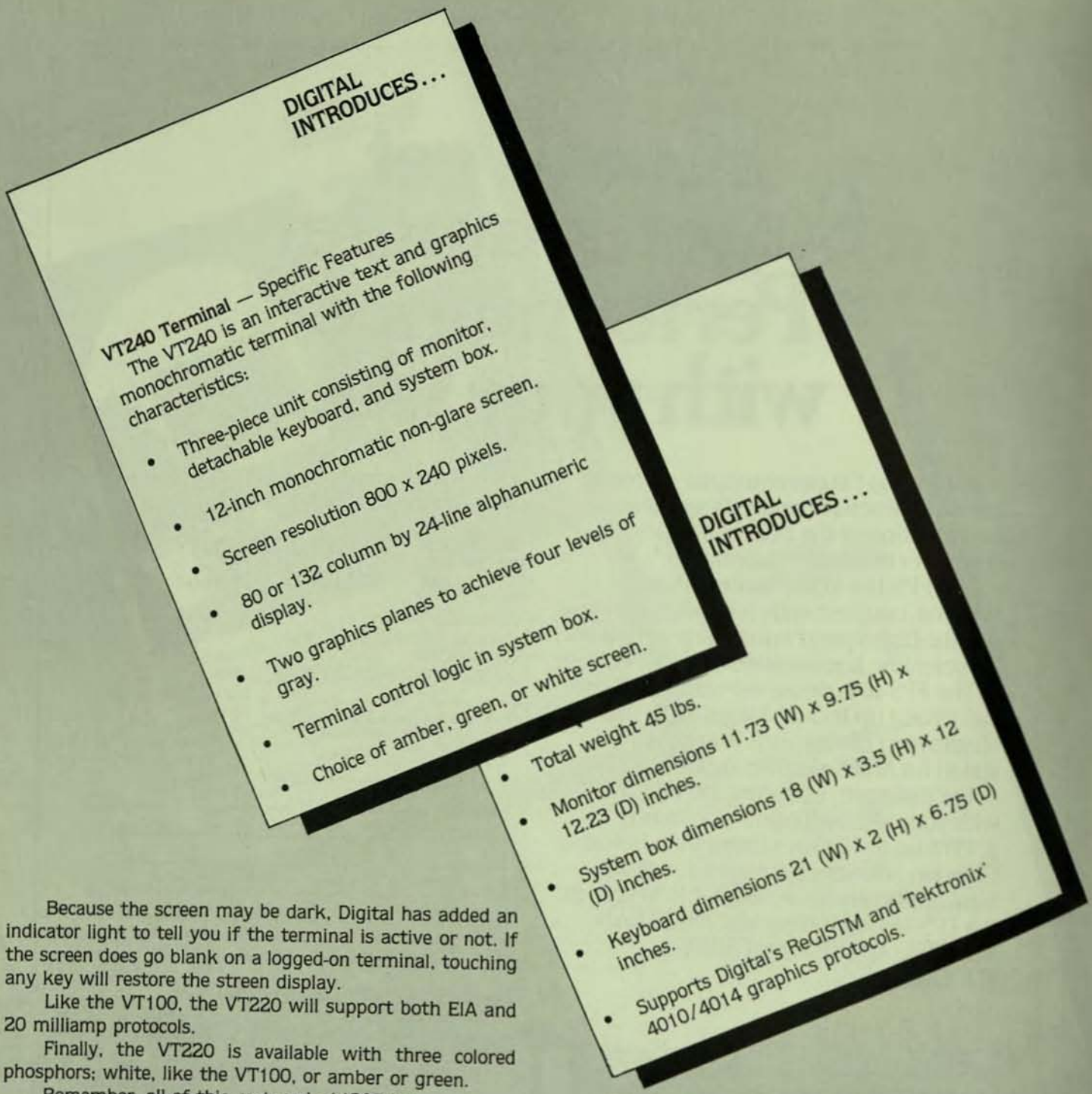
Application _____

*Prices for the FPS-164 start at $300,000.



The VAX 11/780 with the FPS-164.

CIRCLE D152 ON READER CARD



Because the screen may be dark, Digital has added an indicator light to tell you if the terminal is active or not. If the screen does go blank on a logged-on terminal, touching any key will restore the streen display.

Like the VT100, the VT220 will support both EIA and 20 milliamp protocols.

Finally, the VT220 is available with three colored phosphors; white, like the VT100, or amber or green.

Remember, all of this costs only $1295 list.

THE VT240

The VT240 is the monochrome graphics terminal of the VT200 series. It includes all of the VT220 features plus bit-map graphics. It can do everything a VT125 can plus, with eight grey shades available, and graphics down to the pixel level.

The VT240 can display text in multiple fonts and type styles. Text can be scaled to any size, rotated (you can write copy along a diagonal graph line, for example, or vertically in the bars of a chart). The type fonts can be italicized for emphasis.

With an additional printer such as the LA50 attached to the standard printer port, anything on the screen, graphics included, can be printed.

The bit-map option allows you access to 800 horizontal by 240 vertical pixels. With appropriate software (for example, VAX Datatrieve) you can produce bar charts in various grey shades, pie charts, line drawings — and all can be reproduced in hard copy with a single keystroke.

The VT240 supports both Regis and Tektronix 4010/4014 protocols, so it can respond to many available graphics packages such as the Plot 10 Tektronix software.

The terminal operates at a standard 9600 baud, with 19.2 kilobaud available if your system supports it.

An optional, built-in modem is available that will operate at 300 or 1200 baud. The modem is an autodial, autoanswer modem.

A serial input channel is also available to support input from graphics tablets and bar-code wands.

The VT240 lists at $2195 including keyboard.

IS COBOL OBSOLETE? PRO-IV USERS THINK SO.

A totally new approach to application development.

Now you can create fully operational applications in days instead of weeks... and without any programming at all!

PRO-IV is a complete application development and execution system.

Creates applications that are fully transportable between VMS, P/OS, RSTS and RSX*

Generates documentation automatically

A complete library of application packages is available.

For more information, call:
(617) 486-9621

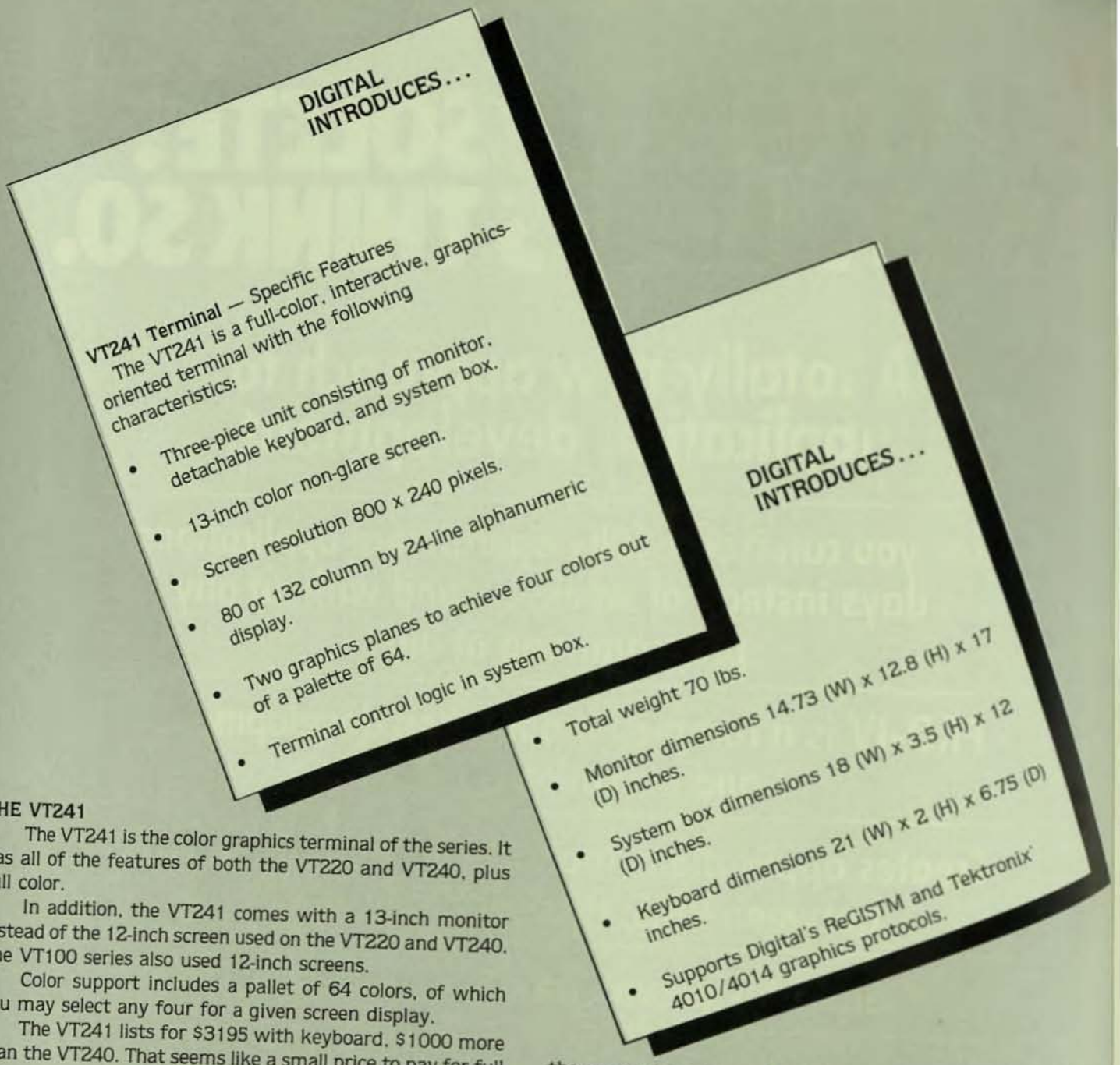
PRO IV INC.

119 Russell Street
P.O. Box 595
Littleton, MA 01460

"THE FOURTH GENERATION APPLICATIONS PROCESSOR"

*UNIX available this fall from Pro-IV.
RSTS, RSX, VMS, and P/OS are trademarks of Digital Equipment Corporation.
UNIX is a trademark of Bell Laboratories.

CIRCLE D102 ON READER CARD



THE VT241

The VT241 is the color graphics terminal of the series. It has all of the features of both the VT220 and VT240, plus full color.

In addition, the VT241 comes with a 13-inch monitor instead of the 12-inch screen used on the VT220 and VT240. The VT100 series also used 12-inch screens.

Color support includes a pallet of 64 colors, of which you may select any four for a given screen display.

The VT241 lists for $3195 with keyboard, $1000 more than the VT240. That seems like a small price to pay for full color, and color displays seem to be the wave of the future.

AVAILABILITY

The Product Manager of the VT200 series told me during the DECUS Symposium in October that the product would not be announced until it was available for delivery in some quantity. It seems that Digital has learned from past mistakes in this respect.

The salespeople who presented the terminals to the New York Local Users Group on November 17, said that they are available now to large OEMs and terminal distributors. For general use the VT200s will be available on a limited basis until February, 1984. From February onward they will be generally available for shipment in any quantity.

WHAT ABOUT THE VT100?

Will the VT100 be discontinued? Will there be a drastic price reduction? Both actions seem reasonable in the light of the aggressive pricing of the VT200 series; there is definitely more bang for less bucks. When I asked a Digital employee at DECTown for his opinion of what the VT200 would do to the VT100 line, he said frankly, "It will kill it." I tend to agree, although it will not happen immediately.

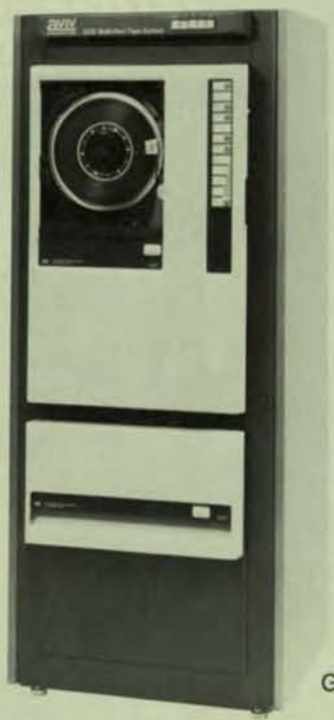
Digital, however, has not yet announced a price reduction on the VT100. Realistically such a reduction may not be possible, at least not enough for DEC to still make a profit on new VT100s. I would certainly expect the used VT100 market to experience a significant price drop.

A PERSONAL OPINION

With an on-site maintenance charge of only $9 per month per unit, Digital seems to have high confidence in the reliability of the new units. If you use a lot of terminals the reduced maintenance costs alone are attractive. And if you are considering adding terminals, you should try out one of the new VT200 series before making any volume purchase.

Protect your Wealth of Information Use Aviv GCR-125 or Aviv GCR-50 Tape System

GCR-50



GCR-125

GCR Format — Perfect Solution to Large Data Base Back-Up

Aviv — the only manufacturer with over four years of field experience with high performance GCR Tri-Density Tape Systems for all DEC and Data General computers, is now shipping the new generation, the GCR-50 Dual Density Tape Systems.

No other company has such extensive experience.

High recording density - 6250 bpi - provides users with the ability to pack approximately 140 mBytes on a single reel. High density coupled with high data integrity and throughput reduces disk back-up time and system overhead. The result...more time for processing.

GCR-125 Tri-Density Tape System records in 6250/1600/800 bpi at 125 ips while **GCR-50 Dual Density** records in 6250/1600 bpi at 50 ips. The transports are manufactured by Storage Technology and integrated with Aviv manufactured tape controllers into a complete, fully-tested system.

Multi-Port Feature — Cost-Saving

Aviv field-proven Multi-Port Feature allows up to four DEC and Data General computers to share a bank of GCR-125 Tri-Density or GCR-50 Dual Density tape drives. This feature optimizes the use of the GCR equipment, saves floor space, and most important, cuts cost. No other company can match this significant feature.

Perfect Match — Efficient and Economical

Aviv is delivering GCR Tape Systems operating at 125 ips or at 50 ips, allowing the user to match tape throughput to system requirements and budget considerations. Aviv GCR Tape Systems are software transparent and operate with VAX 780-750-730, all PDP-11, and all LSI-11 computers including 22 bit addressing and block mode. No other company offers such a wide range of products.

Nationwide Field Service — Complete Solution

All Aviv products are backed by nationwide field service to provide users with installation and on-site maintenance. Aviv Field Service Centers across the country are fully stocked with spare parts.

For solutions to your specific requirements, call Ed Arsenault at (617) 933-1165.

aviv
CORPORATION

Corporate Headquarters, 26 Cummings Park, Woburn, MA 01801
(617) 933-1165 Telex 948539 AVIV WOBN

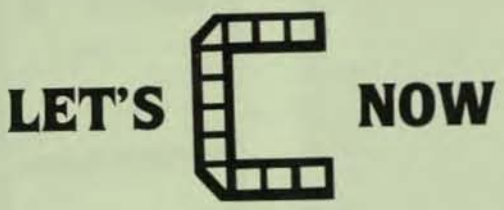
See us at Uniforum Booth #117 and at DEXPO East Booth #1308.

GSA Contract GS-OOK-8401S5567

CIRCLE D105 ON READER CARD

(Editor's Note: Welcom to the world of "C." During the coming months Rex Jaeschke will look at the statements and constructs of the C programming language. Each new construct will be introduced by example with corresponding explanations. Where possible, the examples will be error-free, complete programs and/or subroutines. The reader is encouraged to enter, run and modify these examples as the only realistic way to master any language is to write programs in it, run them and debug them.

All program examples have been taken directly from the manuscript text file and tested using Whitesmiths' C compiler running on a DEC PDP-11/23 with RSX-11M v3.2. This C compiler has the same capabilities across the PDP-11 and VAX-11 series processors and operating systems. Most material presented will be processor and operating system independent although specifics will be mentioned where applicable.)



PART 1, IN THE BEGINNING . . .

By Rex Jaeschke, Washington, D.C.

C is a mid-level or systems language. That is, C fits somewhere between assembler and high-level languages. On the one hand it has many of the capabilities traditionally found only in assembly languages, in that it allows the programmer to get quite close to the host machine architecture and instruction set using constructs such as + + and -- for auto-increment and decrement. Conditional compilation directives are supported, as are bit-field operators and address arithmetic. On the other hand, it supports high-level language constructs such as if-else, while, for and case statements.

C was developed in 1972 at Bell Labs by Dennis Ritchie. The UNIX operating system, another Bell Labs development, was then rewritten in C. UNIX and C (and DEC) have been closely associated ever since. Every UNIX and UNIX-like operating system package includes a C compiler. C was designed by and for professional programmers and is by no means a language for the beginner. Many of C's more powerful capabilities, particularly pointers and structures, require advanced programming experience before they can be fully utilized.

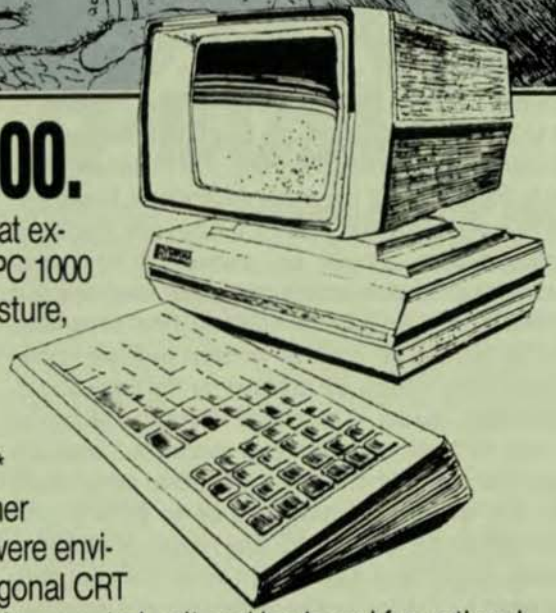
One major reason that software developers use C is the ease with which properly designed programs may be ported across different hardware and system software environments. With rapid development of newer and faster processors, software houses must somehow design programs that will have a life beyond the current systems. The time and cost of rewriting software for new target machines is too prohibitive if the vendor is to retain a reasonable market share. Many PDP and VAX system software houses use C as a tool to assist with program portability. The recursive nature, re-entrant code and ease of implementing b-trees makes C popular, particularly with vendors of file and data base management systems (DBMS) such as Oracle and Ingress.

Using C in itself does not guarantee portability. It is very easy to write machine dependent C code. The decision to make a program portable must be made before the program is designed. There are many different aspects of portability that must be reviewed. Designing portable programs takes care and discipline. Later in this series, we will address portability issues in detail particularly with regard to porting software between PDP-11 and VAX systems, and possibly to and from other machines such as the DEC Rainbow.

The reason that C can provide power similar to assembly language and maintain machine independence at the same time is largely due to the fact that the C language has no provision for any data input and output. "What, no I/O,

THE GPC 1000.

Designed with sealed, airtight enclosures that exceed NEMA 4X and 6P requirements, the GPC 1000 survives in environments where dirt, oil, moisture, heat, corrosives and airborne contaminants are the enemy. Its durable rugged design incorporates advanced technical specifications with smart ergonomics. DEC* compatible, the GPC 1000 outleagues all other video terminals in its ability to perform in severe environments with features like a sealed 14″ diagonal CRT with protective anti-glare filter; separate monitor, control unit and keyboard for optional bench top, wall, shelf or ceiling positioning; true bi-directional auxiliary port for two way communications—and much more—all backed with our one year warranty.



The Severe Environment Video Terminal Engineered To Work Where You Work.

GPC

GAMMA
PRODUCTS CORPORATION
7092 Industrial Loop
Shreveport, LA 71129 USA
(318) 686-1600 Telex 750469

"Tough Products For Industry"

For more information concerning Gamma Products, contact Sharon Russell, Marketing Communications.

*reg. TM-Digital Equipment Corporation emulates VT100 with AVO

CIRCLE D159 ON READER CARD

REAL GRAPHICS POWER FOR THE Q-BUS!

ONE CARD DISPLAYS 256 COLORS. THE OTHER HAS HIGH RESOLUTION "WINDOWING" CAPABILITY. EACH IS AMAZINGLY COMPACT. AND YOU WON'T BELIEVE THE PRICE.

We made them small (just 5.2"x8.9").

We made them powerful.

We made them easy to use (with a complete library of graphics utilities).

And above all, we made them inexpensive, especially when you consider the alternatives.

Since 1979 Peritek has been a leading supplier of high-performance modular display components that plug directly into the Q-bus.* But we've never been more excited than at the introduction of our H-Series products.

Our **Model VCH-Q** dual high card can simultaneously display 256 colors from a palette of 16 million via Peritek's random access look-up table.

We packed lots of features into this board to make applications development easy. For example, under program control you can select alternate color map settings on a frame-by-frame basis so that special effects, animation and so on are possible. You can also double the number of apparent colors by displaying from alternate maps on adjacent pixels. Peritek's extensive software will provide all the support you need.

The **Model VRH-Q** graphics and character video display interface produces both a high resolution 1024 x 1024 monochrome bit-mapped graphic image and an overlaid 64 x 128 alphanumeric display.

Because the character sets are stored in random access memory, you can download an unlimited library of characters and symbols from a host computer.

The VRH-Q can simultaneously display several processes on different parts of the screen, making it particularly suitable for interactive applications. Typical uses include CAD/CAM, robotics and vision systems, and high resolution work stations.

CIRCLE D104 ON READER CARD

EXTRAORDINARY COST EFFECTIVENESS

If you're presently building or using a Q-bus machine, you have a substantial software investment in your system architecture and instruction set. Doesn't it make more sense to simply add a compact plug-in board for $6,000 or less, rather than write more code or pay over $10,000 for a serial device?

Price of the VCH-Q is just $2995, with even lower prices in OEM quantities. The VRH-Q is also $2995. In addition, Peritek offers a variety of accessories and other Q-bus hardware to help you meet your design goals affordably.

For more information call or write Peritek at 5550 Redwood Road, Oakland, CA 94619. (415) 531-6500. TWX: 910-366-2029.

Peritek

*Q-bus is a trademark of Digital Equipment Corporation.

what sort of language is this?" I hear you say. Well, these capabilities are provided by compiler vendor and/or user written routines that are invoked from C programs. This may seem awkward at first, but once you have written a few programs it seems a simple extension to the structured programming approach. It's just like calling MACRO I/O sub-routines from FORTRAN or BASIC-PLUS-2. Most C compiler vendors do not provide source code to their run-time library routines, therefore, I will be presenting my source code versions of many of these commonly supplied functions. Throughout the series we will be developing many simple but useful subroutines. Together they will form a solid foundation for a C programming "tool box."

The definitive reference for C is "The C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie (Prentice-Hall, 1978). Throughout the series, this book will be referred to as K&R. K&R has a tutorial format and encourages the reader to program examples and problems. It assumes prior programming experience and is more of a user's guide than a reference manual. Most C compiler manuals refer to K&R for C language specifics. Any reader seriously interested in evaluating or using C will require a copy of this book which costs about $20 and should be available through most good bookstores. Fortunately, its authors have defined a series of commonly used I/O routines whose naming and calling conventions appear to have been accepted as a standard by most compiler vendors.

C is a simple language with only about 30 keywords. Due to the power of these basic statements and constructs and the fact that the programmer can effectively extend the language by using callable functions and still maintain portability, there is little need for language extensions. However, they do exist in some compilers and vary from one vendor to another. Programmers should be aware of any non-standard conventions in production compilers they use.

Programmers new to C may initially find programs hard to read as almost all code is written in lower case. Lower and upper case letters are treated by the compiler as different characters. In fact, C language keywords MUST be written in lower case. If you have ever wondered what all those special punctuation marks on a VT100 or other ASCII keyboard are for, the answer may well be to write C programs. C uses almost all of them for one reason or another. Note that many ASCII terminals do not have the complete character set used by C. Missing characters usually include |, { : and \, which are all commonly used punctuation marks. Keywords are reserved and the language has been designed as unambiguous, which allows C compilers to be easily written. For example the assignment operator is =, while the equality operator is ==.

C encourages, indeed forces, a structured, modular approach to programming through the use of callable subroutines, called functions. Functions can be separately compiled, with global references and other resolutions being made at TKB or LINK time.

Unlike PASCAL, C is weakly typed. It permits the programmer to do seemingly silly things, particularly in mixed mode arithmetic. This allows the programmer to do powerful tricks and also to make nasty errors. C code can be very tight and obscure to the point of being unreadable and, therefore, unmaintainable. However, with care and discipline, a programmer can generate nicely formatted and easy to read code.

If C is so great, how come everybody is not using it? Well, like other languages, C is not all things to all people. For many applications COBOL, FORTRAN and BASIC do just fine. One aspect of C that discourages commercial programmers from using it is that there is no support for group moves or compares of character strings, arrays and structures. Such capabilities must be provided by external functions, which are generally provided in the compiler vendor I/O library. Unless a programmer really needs the unique capabilities provided by C, he or she is unlikely to use it as a production language. "To C or not to C?" That can be a difficult question. (My apologies to William Shakespeare.)

Program structure

Let's begin by having a look at the basic structure of a C program.

```
/* — smallest.c The smallest possible C program — */

main ()
{
}
```

A C program consists of one or more functions which may occur in any order in a source code file. A program must contain a function called main. Main is a special function name which indicates where the program is to begin executing, that is, its entry point. Main is like the PROGRAM statement in FORTRAN. There is no differentiation between the main program and any subprograms other than their names; they all have the same structure and are referred to as functions.

The parentheses following the function name surround the dummy argument list. The parentheses are required, even if no arguments are expected. The function main can have arguments but in this case, does not. Passing command line arguments to main from MCR, DCL or other CLI will be discussed in a future issue.

Comments are delimited by /* and */ and may occur anywhere a blank, tab or new line can. The pair of braces is required for each function and any executable code for that function must appear inside them. Statements are executed in sequential order unless control or looping statements dictate otherwise. A program terminates when it reaches the closing brace of the function main.

C is a free format language. Spaces, tabs and blank lines can and should be used liberally to improve program readability. They are ignored by the compiler unless they are part of a quoted literal string. A compact form of the above program is

```
main() {}
```

Although both of the above programs will compile and run without error, they have no executable code and therefore do nothing.

REAL GRAPHICS POWER FOR THE Q-BUS!

ONE CARD DISPLAYS 256 COLORS. THE OTHER HAS HIGH RESOLUTION "WINDOWING" CAPABILITY. EACH IS AMAZINGLY COMPACT. AND YOU WON'T BELIEVE THE PRICE.

We made them small (just 5.2"x8.9").

We made them powerful.

We made them easy to use (with a complete library of graphics utilities).

And above all, we made them inexpensive, especially when you consider the alternatives.

Since 1979 Peritek has been a leading supplier of high-performance modular display components that plug directly into the Q-bus.* But we've never been more excited than at the introduction of our H-Series products.

Our **Model VCH-Q** dual high card can simultaneously display 256 colors from a palette of 16 million via Peritek's random access look-up table.

We packed lots of features into this board to make applications development easy. For example, under program control you can select alternate color map settings on a frame-by-frame basis so that special effects, animation and so on are possible. You can also double the number of apparent colors by displaying from alternate maps on adjacent pixels. Peritek's extensive software will provide all the support you need.

The **Model VRH-Q** graphics and character video display interface produces both a high resolution 1024 x 1024 monochrome bit-mapped graphic image and an overlaid 64 x 128 alphanumeric display.

Because the character sets are stored in random access memory, you can download an unlimited library of characters and symbols from a host computer.

The VRH-Q can simultaneously display several processes on different parts of the screen, making it particularly suitable for interactive applications. Typical uses include CAD/CAM, robotics and vision systems, and high resolution work stations.

CIRCLE D104 ON READER CARD

EXTRAORDINARY COST EFFECTIVENESS

If you're presently building or using a Q-bus machine, you have a substantial software investment in your system architecture and instruction set. Doesn't it make more sense to simply add a compact plug-in board for $6,000 or less, rather than write more code or pay over $10,000 for a serial device?

Price of the VCH-Q is just $2995, with even lower prices in OEM quantities. The VRH-Q is also $2995. In addition, Peritek offers a variety of accessories and other Q-bus hardware to help you meet your design goals affordably.

For more information call or write Peritek at 5550 Redwood Road, Oakland, CA 94619. (415) 531-6500. TWX: 910-366-2029.

Peritek

*Q-bus is a trademark of Digital Equipment Corporation.

```
/* — small.c Smallest C program with 2 functions — */

main ()
{
  sub ();
}

/* — sub has no executable code — */

sub ()
{
}
```

C encourages programmers to break large programs up into a number of smaller functions. A source file may contain one or more functions which can be defined in any order. Unlike PASCAL, C function definitions cannot be nested. That is, the function sub must be defined outside of the braces delimiting the function main. A function is invoked by naming it, along with a list of arguments surrounded by parentheses. Arguments are optional, but the delimiting parentheses are not.

Each C statement must be terminated by a semicolon. Invoking a function is considered a statement. The semicolon is a statement terminator, NOT a statement separator as in PASCAL. When function sub terminates at its closing brace, control is passed to the statement immediately following the one which invoked sub.

Function and variable names are traditionally written using lower case letters. Notice how spaces, tabs and blank lines are used to make the program format more pleasing to the eye. Also, the opening and closing braces are aligned to better indicate each function's scope. K&R goes to some pains to recommend a program formatting style. Although their recommended style is widely used and seems to work well, the programmer is encouraged to experiment. Whatever style you adopt, be consistent.

Formatted screen output

```
/* — welcome.c produces output from C — */

main()
{
  printf ("Welcome to the C language. \n")
}
```

C has no input or output statements as part of its language. As we previously indicated, such capabilities are provided as part of a run-time library of functions which are supplied by the compiler vendor and/or user written. Printf is such a library function and prints formatted output to the standard output device which is typically the user's terminal.

The character string to be printed is enclosed in double quotes. The \ n is a special SINGLE character which is C notation for new line. Printf never prints a new line automatically, so it may be invoked multiple times to print an output line a piece at a time. The new line is a line feed character and is typically converted to a CR/LF pair on output. The backslash is an escape character prefix which indicates that the following character is to be treated specially. Other common sequences are \ b for backspace, \ t for tab, \ " for double quotes, \ f for form-feed and \ \ for the backslash itself. This mechanism allows character set independence for these special characters. Some compilers correctly interpret upper case escape sequence letters such as \ N and \ T. There is no guarantee that this is valid on other compilers, therefore, this practice should be avoided.

The above program could be written as

```
main ()
{
  printf ("Welcome ");
  printf ("to the C language.")
  printf ("\n");
}
```

or as

```
main ()
{
  printf (Welcome "); printf ("to the C language."); printf ("\n");
}
```

All three examples are equivalent, however, the first one is more readable and will execute faster as it only involves one call to the printf function. The third example shows that multiple statements may exist on the same source code line provided each is terminated by a semicolon. This practice is discouraged as it generally makes the code less readable. However, there are occasions where it proves useful.

When a function is invoked, all its arguments must be specified on the same source code line. The maximum length of a source code line and a quoted literal string is compiler dependent.

```
/* — variable.cc define and print an integer variable —*/

main ()
{
  int year;

  year = 1983;
  printf ("This year is %d. \n", year);
}
```

"See, right now we don't need anywhere near the performance of an IBM mainframe, so what we'll do when we outgrow our VAX is get another VAX, and another VAX, you know, until we really do need the IBM mainframe, and it'll only cost us three or four times what it would have cost if we'd gotten an ELXSI in the first place."

The ELXSI guarantee: find a computer system that can match our performance and our potential for expansion, and we'll raise our price to match theirs.

The ELXSI 64 Bit Multiprocessor System 6400.
Price. Performance. Potential.
No one can match us. Not now. Not later.

ELXSI

ELXSI, 2334 Lundy Place, San Jose, California 95131. 408/942-1111. Telex: 172320.
TATA-ELXSI, 37 Joo Koon Circle, Jalan Ahmad Ibrahim, Singapore 2262.
Telephone: 2616727/8, 2640011. Telex: RS 34413 TESING.

What do you need for your

system?

Find it at

DEXPO® East 84.

Simply everything for your DEC* computer.

ANNOUNCING THE FIRST CONFERENCE

Save Time, Save Money . . . Mail Today!

DEXPO® East 84

The Fifth National DEC*-Compatible Industry Exposition and Conference

Bayside Exposition Center
Boston
April 3-6, 1984
Tuesday-Friday

☐ Send _____ reduced-rate show tickets plus complete information on exhibits.

☐ Send _____ Conference Programs.

Name _____

Title _____

Company _____

_____

Address _____

_____

City _____

State _____ Zip _____

☐ I am interested in exhibiting my company's products.

Call me at (_____) _____

*DEC is a registered trademark of Digital Equipment Corporation

C D E

and top
d confer-
agement
ns on Of-
r Graph-
nference
nt DEC-
ast 84.

rd. Or write
Hightstown

ter

What do you need for your

DEXPO®

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 66 PRINCETON JCT., NJ 08550

POSTAGE WILL BE PAID BY ADDRESSEE

Expoconsul International, Inc.
55 Princeton-Hightstown Road
Princeton Junction, NJ 08550

Find it at

DEXPO® East 84.

Simply everything for your DEC* computer.

Like most other Digital customers, you're probably always looking for new ways to get more out of your computer system. Today, that demands the very latest DEC-compatible products and services. And for more than 10,000 owners, managers, users and dealers of DEC computers, that demands a visit to DEXPO East 84.

Here's why you should be there, too.

New DEC-compatibles are being introduced almost every day. Software. Hardware. Services and supplies. Even complete systems. You need to explore, examine and evaluate *all* your choices. In one place. At one time. Without distractions. And you can only do it at DEXPO East 84.

The most complete exhibition of DEC-compatibles yet.

Some 400 vendors will be at DEXPO East 84 to demonstrate thousands of products and services that work with your DEC equipment. You'll find a spectrum of software for your Rainbow*. Plenty of peripherals for your PDP-11*. All the facts on compatibles for your VAX* . . . and *more* . . . for every DEC computer.

*Registered trademark of Digital Equipment Corporation

ANNOUNCING THE FIRST CONFERENCE FOR DEC-USING EXECUTIVES

For the first time, DP managers, MIS directors and top corporate executives can attend a DEC-oriented conference covering today's crucial information management issues. The four-day conference features sessions on Office Automation, Personal Computers, Computer Graphics, Communications, Security and more. The conference is sponsored by IRUS, the national independent DEC-user association, in cooperation with DEXPO East 84.

For complete information mail the attached reply card. Or write to Expoconsul International, Inc., Dept. DE, 55 Princeton-Hightstown Road, Princeton Junction, NJ 08550.

DEXPO® East 84

The Fifth National DEC*-Compatible Industry Exposition and Conference

Bayside Exposition Center
Boston
April 3-6, 1984
Tuesday-Friday

Variable names may be any length but C only recognizes the first eight characters. Some compilers recognize longer names. Avoid this language extension if portability is of concern. Names are made up of letters and digits and the first character must be a letter. The underscore "_" is allowed and is considered a letter. Upper and lower case characters are different, for example year and YEAR are two different variables. Traditionally, variable names are written in lower case.

Variables must be explicitly declared. This "evil" is necessary if one is to overcome the nasty errors possible with default tying as available with FORTRAN and BASIC. On any significant size programming project, explicit typing becomes more an asset than a liability. A declaration consists of a type followed by a list of variables which have that type, such as

```
int start,end,inc;
```

Here, year is declared to be of type integer. The size of the number that can be stored in an int variable is machine dependent as are most C data types. Other data types are char (character), short (short integer), long (long integer), unsigned integer, float (single precision floating point) and double (double precision floating point). Specific storage requirements for DEC systems are as follows:

	PDP-11	VAX
char	8 bits	8 bits
short int	16 bits	16 bits
int	16 bits	32 bits
long int	32 bits	32 bits
unsigned	16 bits	32 bits
float	32 bits	32 bits
double	64 bits	64 bits

On some machines (i.e., PDP-11), int and short are the same, while on others (i.e., VAX) int and long are equivalent. On some non-DEC machines it is possible for short, long and int to have the same storage capacity. Float and double may also have the same precision. A char variable may be 6, 7, 8 or more bits. Obviously these factors are important to programmers writing code to be portable across different machine architectures. Fortunately, there are techniques available to implement portable code across machines with differing word sizes. These will be discussed in future articles.

Variables are assigned values using the symbol "=". The value assigned may be a constant, the value of another variable or the result of a function, or an expression involving any or all of these.

The printf function call has two arguments this time, a character string and an integer variable year. The string serves as a format or edit mask and contains a special sequence %d which causes the first argument after the string to be interpreted and printed as a decimal integer. Other masking sequences exist for each data type and the masks present in the string determine the number and type of arguments that are expected after the first argument. The value of year will be inserted into the text string when it is printed, followed by a new line as indicated by the \n. The output generated is: This year is 1983.

Note that year is defined inside the braces of main, an area which is reserved for executable code. The definition int year; can be considered as executable, as storage for the variable year is allocated at run-time, and NOT at compile time. Variables declared within a function are local to that function and are not accessible to other functions. They are created EACH time their parent function is invoked and disappear when that function is exited. For this reason they are known as automatic variables.

Storage for automatic variables is typically allocated on the hardware stack. Therefore, a large number of concurrent automatic variables, in particular arrays and structures may cause stack overflow. The size of the stack may be determined by the compiler and/or linker used. The IAS/RSX Task Builder has a default stack size allocation of 256 words. Use the TBK STACK option to change this. On VAX/VMS stacksize allocated by Linker can also be changed using the STACK option.

Space for automatic variables is generally reserved by adjusting the stack pointer. Therefore, these variables should be explicitly initialized else they will contain garbage.

Let's look at a set of variable declarations.

```
sub ()
{
  char c1,c2;
  short s;
  int i;
  long l;
  unsigned u;
  float f;
  double ;
}
```

On the PDP-11, 24 bytes are reserved on the stack for the eight automatic variables, one for each char, two each for short, int and unsigned, four each for long and float and eight for double. On the VAX, 28 bytes are reserved with two extra for each long and unsigned variable. When function sub terminates, this storage space is released by adjusting the stack pointer to its value on input. If the declaration for the char variable c2 is omitted on a PDP system, the compiler still reserves 24 bytes. That is, it forces word (even byte) alignment on the stack to avoid our friend, the Odd Address Trap.

I hope this introduction has whet your appetite for more information about the C language. Next time we will take a look at some of the looping and branching constructs and will introduce symbolic constants and character I/O. Future installments will be more "meaty," particularly in reference to code generation efficiency and will include machine code examples as appropriate. Beginners or non-technical readers are encouraged to skip these technical considerations until they are more proficient at the language.

C you next time.

Note: Rex Jaeschke is a consultant, educator and writer based in the Washington, D.C. area. His current interests include languages (in particular C and PASCAL), and the user of microcomputers for distributed data processing, communications and computer aided education.

A unique Word Processor
for **DEC™... by Quadrex**

Finally see it on the screen

- Exactly as it will appear on the print-out.
- No confusing codes on screen to slow you down.
- Ease of operation — especially useful for beginners.
- The Quadrex QX-8000 WP Retrofit for VT100™... offers a wealth of word processing capabilities.

- Proportional spacing on the screen.
- Left and right justification on the screen.
- Subscripts and superscripts on the screen (without codes).
- **Bold...** underline... center... multiple level indents... all on the screen without special codes.
- On line HELP facility instantly available on the screen.
- Numerous new features and functional advances are engineered into the QX-8000 Retrofit.

Dealers' inquiries invited.

For more information call 1-800-227-3800
Ask for your Quadrex representative

DEC™ and VT-100™ are trademarks of Digital Equipment Corporation.

Quadrex Word Processing Systems Inc.

Head Office
4770 Kent Avenue, Suite 316
Montreal, Quebec H3W 1H2
514-731-7551

Boston Office
50 Milk Street, Suite 1500
Boston, MA 02109
617-451-6509
telex: 951131 HQ BSN

CIRCLE 0169 ON READER CARD

Computer Bookshelf

Legal Care for Your Software
By Daniel Remer
Nolo Press, 1983
Berkley, CA
247 Pages, $19.95

If you are involved on any level with software development, "Legal Care for Your Software," is required reading. This modestly priced paperback is an excellent primer on the legal rules relating to the development, protection, and marketing of software.

Remer, who is an attorney and software enthusiast, asserts that the best way to avoid legal pitfalls is to prevent problems from occurring in the first place. He makes his reader very aware that full legal protection begins with your having a bit of knowledge about what legal tools are available and what kinds of problems you might encounter. Written in plain English, Remer defines the legal doctrines and devices that affect software and relates these to practical problem situations.

For example, what if you misspell your name or insert an incorrect name on your copyright notice? Under the Copyright Act of 1976, your notice is still valid. You would have five years to correct this common defect.

Each chapter of Remer's book is dedicated to a discussion of a specific area of law: trade secrets, copyrights, contracts, trademarks, patents, and more. Within each chapter, Remer covers his topic thoroughly but he is not overly technical. In the appendices there are sample contracts — the perforated paper makes them easy to tear out and use.

The book presents a plethora of self-help legal prevention measures and is a treasure for any serious software developer.

VAX-11 Assembly Language Programming
By Sara Baase
New Jersey
407 Pages

This is one of the rare VAX books on the market which is not published by DEC.

It is extremely well written. It contains information which is useful to the novice as well as the experienced assembly language programmer. A quick look at the table of contents illustrates how much useful information this book contains — Chapter 1: Machine Organization; Chapter 5: Simple I/O Macros; Chapter 12: Bit and Bit Field Operations; and my favorite, Chapter 15: Input and Output Using RMS.

Each chapter contains many good examples which help the reader understand the text. The examples are generally short, clear, and well presented.

At the end of each chapter a number of exercises are included which illustrate the important points. Unfortunately, answers are only provided for about one-third of the questions. Spending over an hour to solve the exercises for one chapter and not being able to determine if they were correct was extremely annoying.

After reading this book, I had a good understanding of VAX assembly language programming, as well as with the VAX architecture in general. I would highly recommend it to anyone with a desire to explore either of these subjects.

By Richard Lowe
and Amy Parry

(**Editor's Note:** For assistance in reviewing the new "BASIC For The DEC10" we asked Anthony Stracciolini for his expert opinion. Tony is the manager of The Medical School Computer Facility of the University of Pennsylvania, Philadelphia, PA.)

BASIC For The DEC10
By Irmtraud Seeberg & Cynthia Ma
Engineering Press, Inc. 1982

This manual was designed to direct a student through a step-by-step approach to learning BASIC on the DEC10 system.

In the beginning, the student is introduced to the DEC10 system plus all of the commands necessary to operate the BASIC system. Subsequent chapters lead the student through BASIC terms and simple expressions, computations, branching, looping, subscripted variables, input-output control, functions, subroutines, and finally, file processing. Each chapter contains examples with flow charts, listings, and responses that show the student what he should receive if the problem was properly entered into the computer. Each chapter concludes with a set of exercises and programming problems designed to reinforce the objectives the chapter, and allow the student to have an interactive learning experience with the BASIC system.

While I believe that the authors have accomplished their goal of introducing the student to BASIC on the DEC10 system, I question the wisdom of including a section on MAT statements for matrix manipulation in Chapter Six on subscripted variables. The original Dartmouth version of BASIC included this aspect of matrix operations however, in today's world of microcomputers, MAT operations are considered to be an unnecessary expansion of the BASIC software. Further, matrix operations are easily programmed through the use of "FOR" loops. Broader discussions and examples concerning matrix algebra would be more suited.

In the chapter on file processing, the authors have excluded the concept of random access files. In today's world of databases and database management systems, the subject of random access file processing is critical. A discussion of the concept of random access files, and how it is implemented in BASIC on the DEC10 should have been included.

This manual can be considered a sufficient starting point for a student desiring to learn BASIC on the DEC10. With BASIC as a background, the novice programmer can then move to any other computer system that offers a BASIC system. ■

NIFTY THINGS TO DO WITH

RSX INDIRECT COMMAND FILES

By Allen A. Watson, The Record, Hackensack, NJ

Here are more "Nifty Things To Do with RSX Indirect Command Files," in our continuing series on hints, kinks, and tricks using RSX Indirect Command Files. Remember that the articles assume you know how to write the basic Indirect Command Files; i.e., combining compiles and task builds, setting up a job and prompting the user for options, or running a whole series of PIP copies or deletes.

NESTED COMMAND FILES AND PARAMETERS

As with other programming languages, Indirect can have external subroutines, in the form of nested or chained command files. Standard Indirect allows up to four levels of nesting; this can be decreased (but not increased) through another task build option in [1,20]ICMBLD.BLD. Nested command files are invoked by saying "@filename" from within a command file.

When Indirect encounters a ".EXIT" statement in a secondary (nested) command file it returns to the called command file where it left off; if Indirect encounters a ".STOP" command, it exits the entire command file sequence, back to MCR or DCL.

Passing Parameters to Command Files

When a command file is invoked — whether from MCR or from within another command file — the entire command line (including "filespec") is put into the variable COMMAN. Indirect then automatically parses any parameters following "filespec" into the reserved local symbols P1 through P9, using a single blank as the parameter separator. (Two or more successive blanks will result in null parameters.) The symbol P0 is assigned the string "filespec" (I have never found any use for this; when would a command file not already know its own name? It would be better if what got passed was the name of the caller.)

It is not well documented that alphabetic strings passed as parameters are always translated to upper case before being stored in the parameter variables. This happens whether or not you enclose them in quotes; in fact, the quotes go along as part of the parameter. The parsing is done strictly on separating blanks. There is no way I have found to pass lower case letters to command files in a command line.

Finally (and this also is undocumented), the special symbol <STRLEN> is set to the number of parameters received, including P0. Thus, when "@TEST ONE" is executed, when you enter TEST.CMD, <STRLEN> will be set to 2.

Note that all of these symbols are local; that is, they have these values only in the one command file. If A.CMD calls B.CMD with a parameter list, B.CMD receives its own set of parameters, and when Indirect returns to A.CMD, the parameter symbols there will have retained whatever value they had before the call.

Checking for Required Parameters

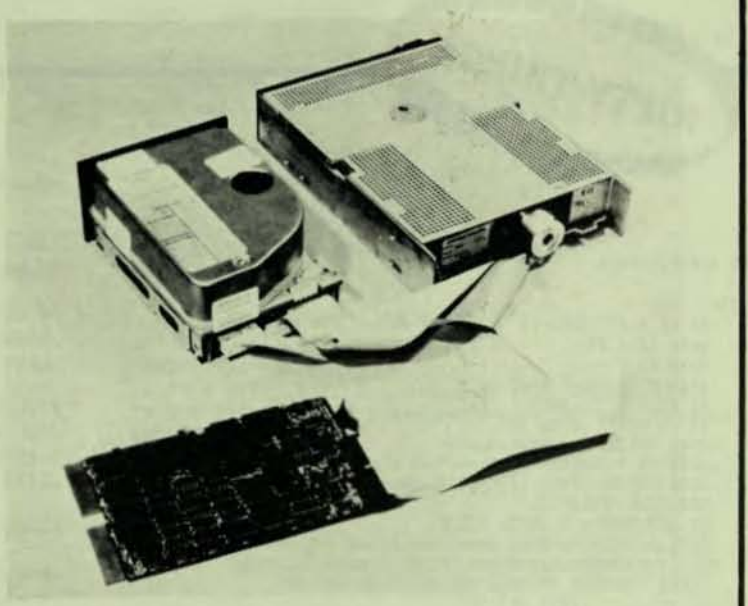
The fact that P1 through P9 are assigned null strings if no parameter is given (they are not undefined) makes it easy to test whether or not a parameter was given. The command file in Figure A illustrates how to check for parameters and prompt if required ones are omitted.

You could also test for parameters by using the fact that <STRLEN> is set to the parameter count (including P0). I would start by saving the count in another symbol, since <STRLEN> gets reset by any string operation:

```
.SETN PARMS <STRLEN>
.IF PARMS < = 1 ; Tests to see if P1 was not given
.IF PARMS < = 2 ; Tests to see if P2 was not given
.IF PARMS > 3 ; Tests to see if P3 was given
.; ETCETERA
```

The WDC11 series...

The FIRST family of Winchester/ Floppy controllers for the LSI-11/Q-Bus



WDC11 connected to a VERTEX V 170 5¼" Winchester disk drive (56.6 Mb) and a shugart SA810 8" floppy disk drive. (Notice the simple interconnect.)

NEW FEATURES: • AUTO CONFIGURATION • TRANSPARENT BAD BLOCK REASSIGNMENT • 70% FASTER PERFORMANCE • 22 BIT ADDRESSING ON WINCHESTER AND FLOPPY • SUPPORTS REMOVABLE CARTRIDGE WINCHESTERS

Unsurpassed Freedom of Choice

Choose the disk drive sizes that are right for YOUR application: The WDC11 controls 8" and 5.25" drives, Winchester and floppy.

The DEC emulations that YOU need: RK05, RL01/02, RP02/03 for the Winchester and RX02/03 for the floppy. All emulations provide 22-bit addressing (RLV12-compatible).

The disk drive vendors that YOU want to use: Seagate, Shugart, Disctron (RMS) Vertex, ATASI, DMA, Quantum, Tandon, Computer Memories, Rodime, Ampex, BASF, Memorex, Micro Peripherals. Any drive with a standard interface ST506, SA1000, SA800/850, or TM100-4 can be controlled by the WDC11.

The controller form-factor YOU want: A single, dual-width card includes the Winchester controller, the floppy controller and an intelligent bootstrap. There is no external formatter board that you need to mount and supply power to. The WDC11 requires only 2.7A of +5V.

The configurations YOU need: The WDC11 is available as a single board, a pre-packaged disk subsystem, or a complete computer system. Systems come in rack-mountable or table top enclosures.

The experience YOU are looking for: Andromeda has been shipping WDC11's since the summer of 1981.

The growth potential that YOU require: The WDC11 is easily adaptable to new, higher capacity, higher performance disk drives as they become available. New emulations and data formats can be handled in a similar fashion.

DEC, LSI-11, Q-BUS, RK05, RL01/02, RP02/03 and RX02/03 are trademarks of the Digital Equipment Corp.

ANDROMEDA SYSTEMS INC.

9000 ETON AVENUE • CANOGA PARK, CALIFORNIA 91304 • (213) 709-7600 • TWX (910) 494-1248

CIRCLE D185 ON READER CARD

```
.ENABLE SUBSTITUTION

.OPEN TI:
        .IF P1 = "" .GOTO 50  ! IF NOT GO ASK FOR FILE NAME
        .sets file P1
        .GOTO 60
.50:    .ASKS FILE Print what file:
         IF FILE = "" .EXIT
.60:    .IF P2 NE "" .GOTO 70
        .ASKN COPIES How many copies
        .SETS P2 "'COPIES'" ! These two lines show how to
.70:    .SETN COPIES 'P2'   ! convert numeric to string
        .TESTFILE 'FILE'
        .IF <FILERR> = 1 .GOTO GOTIT
        .DATA No file by that name; was it one of these?
        @LB:[1,2]INDSYS/LB:INDPRF 'FILE' ! Parse filespec
        .PARSE <EXSTRI> "" NODE DEV UIC NAME EXT VER
        .IF DEV = "" .SETS DEV "SY"
        .SETS TRIAL NODE+DEV+":"+UIC+NAME[1:3]+"*"
        SRD 'TRIAL'
        .GOTO 50
        .ENABLE SUBSTITUTION
.GOTIT: .data Printing 'FILE' on Diablo printer, 'COPIES' copies.
.ask YN Is this correct?
 iff YN  goto 50
 SETF SECOND
.SETF DCL
.IF <CLI> EQ "DCL" .SETT DCL
.IFT DCL SET TERM MCR
        .ift SECOND .goto TYPE
        .ASKS TYP Set Diablo to top of form, press Return.
.TYPE:  PIP TT22:=LB:[1,5]FF.TXT ! Output a single form feed
        .ENABLE QUIET
        PIP tt22:='FILE'
        .sett SECOND
        .DEC COPIES
        .IF COPIES > ^ .GOTO TYPE
        PIP TT22:=LB:[1.5]FF.TXT
        .sets P1 ""
        .SETS P2 ""
        .ASK TRY More?
        .IFT TRY .GOTO 50
        .IFT DCL SET /DCL=TI:
```

You could also test for parameters by using the fact that <STRLEN> is set to the parameter count (including P0). I would start by saving the count in another symbol, since <STRLEN> gets reset by any string operation:

```
.SETN PARMS <STRLEN>
.IF PARMS <= 1 ; Tests to see if P1 was not given
.IF PARMS <= 2 ; Tests to see if P2 was not given
 IF PARMS > 3  ; Tests to see if P3 was given
.; ETCETERA
```

FIGURE A.

PASSING INFORMATION USING GLOBAL SYSTEMS

Indirect also allows the definition of global symbols that are known to all the levels of nested command files during execution, similar to FORTRAN COMMON.

Enabling Globals

Each command file that is to have access to the global symbols must contain the line ".ENABLE GLOBAL" to allow reference to the symbols. When global symbols are used, it is not necessary to pass information in parameters.

A global symbol must begin with a dollar sign; any symbol beginning with a dollar sign is a global.

Figure B is a simple exercise showing global symbols:

```
FILEA.CMD:

.enable substitution
.enable global
.askn $NUM Enter global number
 askn NUM Enter local number
@FILEB
; Back in FILEA
; $NUM = '$NUM', NUM = 'NUM'
.EXIT

FILEB.CMD:

.enable substitution
.enable global
.ifdf $NUM ; In FILEB, $NUM is known to be '$NUM'
.setn $NUM 100
.setn NUM 0
; In FILEB the local NUM is set to 0
.EXIT
```

FIGURE B.

If you run these files you will see that the global, $NUM, which is altered in FILEB, is also altered when we return to FILEA, whereas the local symbol NUM retains its original value.

Making Local Symbols Global

By using a switch, "/LO", you can cause all of a command file's local symbols to be known to another command file. For example, in the above example, if the call to FILEB were changed to "@FILEB/LO", then NUM would be treated exactly as if it were a global symbol. When FILEB altered it, it would be altered for FILEA also. In effect, the /LO switch makes all local symbols into globals for that one command file invocation only.

Returning Values from Nested Command Files

Global symbols (or the /LO switch) can, as has just been seen, be used to return values from a nested command file to the caller. Besides global symbols, there are two means of returning information from a command file to its caller: the special symbols "<EXSTAT>" and "<EXSTRI>". These predefined symbols are global in scope even if GLOBAL has not been enabled.

Exit status (EXSTAT) is set by the ".EXIT" statement. It is normally set to a numeric value of 1, representing success. (Exit status is also set by any task executed from a command file that issues status upon exit, such as PIP, MAC, TKB, or any compiler.) You can cause status to be set to another value indicating an error condition so that, upon return from the nested file, the caller can test it and take appropriate action. For example, a nested command file can exit with ".EXIT 2". The calling file can then test this upon return:

```
@subfile
.IF <EXSTAT> = 2 .GOTO ERR2
```

The EXSTRI symbol is a string variable you can set just prior to exiting to pass additional information back to the caller. Simply use

.SETS <EXSTRI> "Text string of information for caller"

.EXIT

The caller can then do whatever necessary with the information passed back.

Figure C is an example of a command file DEVONL.CMD that passes back to the caller a list of all devices of a certain type that are on-line. The caller can then parse EXSTRI to isolate the device names. DEVONL uses the powerful ".TESTDEVICE" command. Just by changing the ".TEST" line to refer to "MOU" instead of "ONL", you could create a command file to test which devices of a type are mounted for Files-11 access.

```
DEVONL.CMD:
.enable substitution
.IF P1 = "" .EXIT 4      ! Error return, no parameter
.SETN N 0
.SETS BUF ""
.SETN MAX 7
.LOOP:
   .SETS DEV P1+"'N'"+":"
         .TESTDEVICE 'DEV'
         .TEST <EXSTRI> "ONL"
         .IF <STRLEN> = 0 .GOTO SKIP
         .SETS BUF BUF+DEV+" "
.SKIP:
   .INC N
         .IF N <= MAX .GOTO LOOP
.SETS <EXSTRI> BUF
; EXSTRI is '<EXSTRI>'
.IF P1 = "TT" .EXIT 2 ! Warning, list incomplete
.IF BUF = "" .EXIT 5 ! Severe error, no such devices.
.EXIT
```

FIGURE C.

FROM THE PUBLISHERS...

. . . continued from page 4

Lean and Mean, the new DEC machine. That was the description of the new corporate structure after reorganization in Maynard.

Promises, promises.

Instead we got total confusion. Orders were lost, shipping estimates were unavailable and the company surprised the financial world by not knowing what was happening to itself.

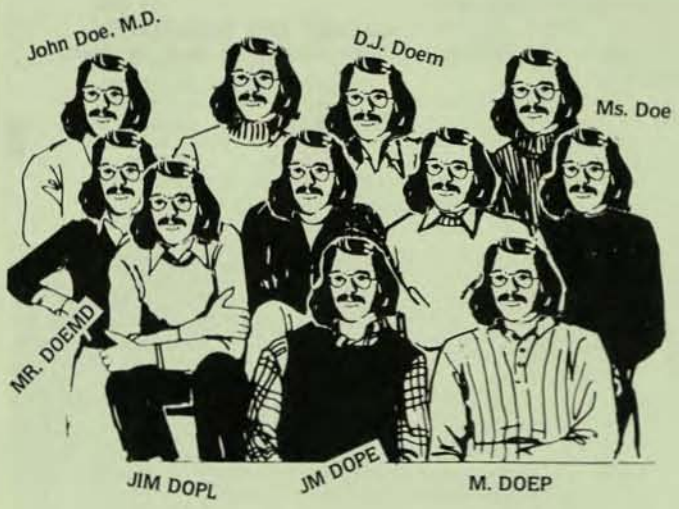
A promise fulfilled was my Rainbow 100. I haven't had this much fun with a computer since I first used a PDP-6 in 1967. FORTRAN, BASIC, spreadsheets, word processors, data bases, games, communications and more. Fun, useful and growing everyday. But if the Rainbow was a promise made good, the Professional 350 was one not kept. Where is the telephone management system that looked like it had possibilities? Why isn't there a real BASIC (commercial kind) in the toolkit? And why do those menus keep software suppliers from moving their PDP-11 software to the Pro? Things are finally beginning to happen with the Pro and maybe next year they can keep some promises too.

The really successful people in this world are not just the ones with "promise," but the ones that go out and "get it done." That goes for companies as well as people. The time has come for DEC to go forward and do it: "GET IT DONE."

THE MAILING LIST AND I

By Frank R. Borger

Michael Reese Hospital and Medical Center, Chicago, IL



WILL THE REAL JOHN DOE PLEASE STAND UP!

It doesn't make sense to me. I work with computers, I program them, I fix them when they break. The ones around me like me, and I like them, but the ones far away that have me on their mailing lists just love to do things to my address. I must admit, my full address is a handful:

Frank R. Borger, Physicist
Instrumentation Division Head
Department of Medical Physics
Mail Stop: MRMEZZ
Michael Reese Hospital and Medical Center
29th Street and Ellis Avenue
Chicago, IL, 60616

That's a bit much for the average mailing list software to handle, but ever since I got on my first mailing list at work, I've had more than my share of trouble.

My first problem was with an engineers book club. I wrote them with what I thought was a reasonable request, that they include my department or my mail stop in my address. (I was often getting the books before I got the card to send in saying don't send the book.) After several attempts (including punching one letter on paper tape, and sending just the tape to the publisher's data processing department), I gave up and quit the club. I did learn one thing: it's easy to get your name on a list, but hard to get it off or change it.

Since then, things have gone from bad to worse. I quickly discovered (by filling out a few request cards at the back of magazines) that people had trouble reading my printing. "I'll solve that," I said, and wrote a program to print my return address on peel-off gummed labels on our LA36. Surprise! The LA36 lower case "g" looks very much like an "s." As a result, I'm on several mailing lists as "BORSER." (I've even had correct mailing labels hand "uncorrected" to say "BORSER.")

I have also seen "Frank R." be reduced to "F. R." and then "F R" and finally "FR" until some brilliant program decided that "FR" was an abbreviation for "Father." Some computer somewhere thinks I'm a priest!!! (At Chicago's largest Jewish hospital, no less!)

Computers (or people) also seem to have a large problem with abbreviations for "PHYSICIST," and our mail room, too, was sometimes confused depending on the abbreviation used. I'm used to getting mail with the remark "NOT A PHYSICIAN" scrawled on the front, but I still haven't gotten over the letter I received with the remark, "NOT IN PSYCHIATRY" noted at the top, and with an even more biting "BUT SHOULD BE" added below.

Computers also have a tendency to try to promote me beyond my real position. Although I'm only "HEAD" of a small (3-person) "DIVISION" of one "DEPARTMENT" of the hospital, mailing list programs must be based toward certain words such as "HEAD." On occasion I appear to be either "HEAD" or "PRESIDENT" of this institution. (I hope the real president doesn't take offense, I'm not really gunning for his job.)

I've seen "DEPARTMENT OF MEDICAL PHYSICS" get clobbered in various ways ("DEPT OF MEDICA," "MEDICA MED," etc.) but I'm amazed at the letter writing software that managed to take our mail stop, "MR MEZZ" (it stands for "MAIN REESE, MEZZANINE") as a name, and started out the letter, "DEAR MR MEZZ:"

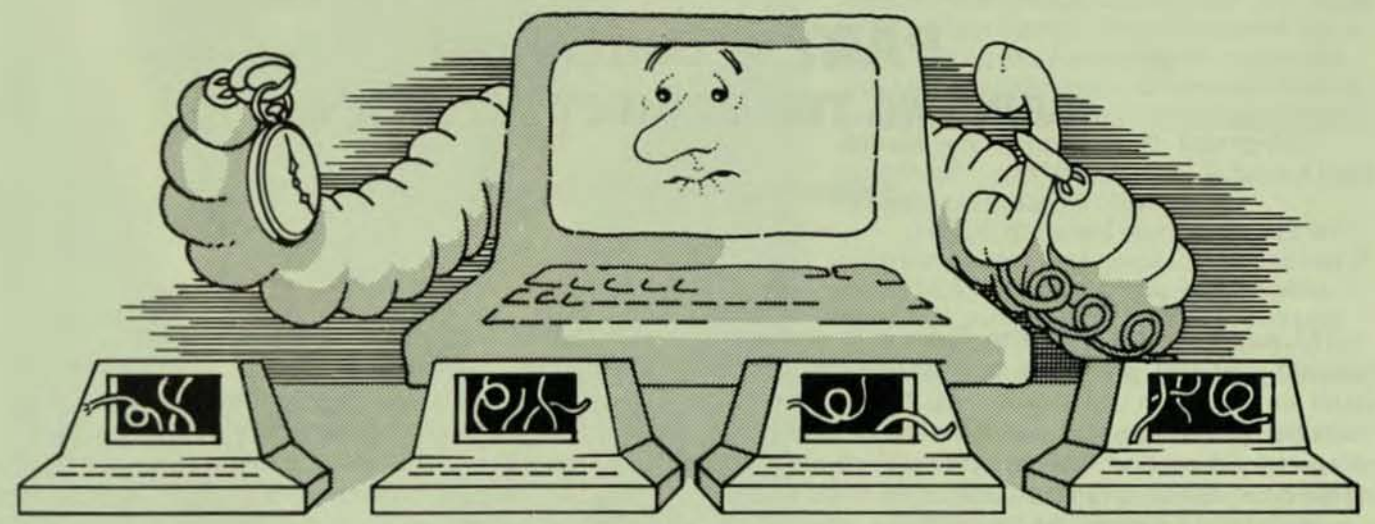
Even the hospital's name has taken its licks. I can understand a shortening to "MICHAEL REESE HOSP," but I've gotten mail addressed to "MICHAEL REESE SHOP," and to the "NED CENTER" (where all the guys named "Ned" meet, I guess). The worst cut was the letter addressed to "MICHAEL R. HOSP" which began "DEAR MISTER HOSP:" and later got even chummier. . . . "YES MICHAEL, YOU TOO . . ." somehow the idea of a computer getting chummy with a hospital doesn't seem to work. Where can they be seen together?

Maybe things will get better, but I'm not holding my breath. That book club I dropped out of a few years ago had a cryptic code "APD" next to my last name. You guessed it. I just got a letter from Time, Inc. asking, "HOW CAN F. APD get the Hammond World Atlas FREE?"

Some days I think that I should have become a musician as my father wanted.

THE DRS DECISION: PART 4

IS YOUR DATA BASE SYSTEM "OEM-ER" FRIENDLY?



If your turnkey system doesn't have a strong DBMS to lean on, then you might be losing potential buyers. When software buyers evaluate a turnkey system today, they base much of their decision on the strength of the DBMS foundation that supports the product. They know that a good DBMS foundation provides power and data integrity without loss of response, flexibility or security. Also, they must be certain that as the DBMS is enhanced, their application will benefit.

DRS, the complete data base management system from Advanced Data Management, offers all of these benefits to software buyers, plus even more for the *OEM-er*. Beyond providing a data base system with its own query language, screen-oriented transaction processor, sophisticated report writer and host-language interface, DRS features PDP-11, VAX and IBM versions, the ability to add user-defined commands to the query language, execute-only versions, and one of the most extensive debugging facilities in the industry today. DRS is designed to help you, the system builder, deliver the goods!

Also, we realize how important strong service and support are to you. DRS has a comprehensive maintenance program that includes complete train—ing, Hotline service, and a unique on-line trouble-shooting response system. And you will find that we have one of the most *OEM-er friendly* discount schedules in the business.

Let us help you get the "word" on how over 200 users have managed some very complex applications using DRS; ...often without a programmer!

ADVANCED DATA MANAGEMENT
15-17 Main Street
Kingston, NJ 08528
(609) 799-4600

CIRCLE D189 ON READER CARD

Send me the word!

Note: In the last issue (Vol. 2 No. 6), the elements required to develop the Functional Specifications were defined. This article will continue the Definition Phase elements and detail required to structure the information within the Functional Spec document.

SYSTEM METHODOLOGY
PART 4 Continued
BRINGING THE PROJECT INTO VIEW

By Robert J. Walsh, Winter Park, FL

The purpose of the Definition Phase is to produce a document that will define the new architecture of the system to the user in user terminology. The document, or Functional Specification produced at the end of the phase is the operational contract between the user, management and the data processing function. In addition, it will be used as input to the next phase, the Design Phase, where it will be reformatted into a programmer readable document. The user talks a different language than the technical staff and should not be expected to decipher a programmer specification into an acceptable English translation.

As each phase is completed, more and more resources will have been expended and will be committed. It is, therefore, important to include as much detail as possible into the Functional Spec to allow management to determine if the project is on course, within budget and still worth pursuing.

Incomplete specs will only cause management to assume something is going to be done in a certain way when, in effect, it may have been eliminated altogether! Once management becomes familiar with a standardized layout of a corporate Functional Spec, they will be able to pick and choose which sections are of interest to them on any given project and review those sections in detail. For example, the sales manager will be looking for report detail while the computer operation's manager will be looking for new or additional hardware requirements, including scheduling factors.

The other specification elements will be used as supplement or for referral purposes as needed. In any case, much of the information developed during this phase will serve as a project directional and developmental checklist and, unless severe changes are made, can be used as part of or as direct input to the Design Phase Document. The number of items (or segments) required in a Functional Spec will vary from corporation to corporation.

FUNCTIONAL SPECIFICATION CONTENTS AND STRUCTURE

The Functional Specification is to the user what an architectural drawing is to the builder. It specifies the structural requirements, defines parameters, identifies certain materials required to complete the project and presents a list of regulations, restraints and constraints. It allows the "builder" to grasp the overall impact of the new product and plan his resources and time accordingly.

1. THE TITLE PAGE — Should contain the following:
   - The name and identification (number) of the project.
   - The analyst(s) name and title.
   - The date the report was produced.

2. THE TABLE OF CONTENTS — contains the major headings and page numbers of each section. In addition, it should contain a list of subtopics, also cross-referenced to a page number for easy referral purposes. This is especially handy during a formal presentation.

3. THE EXECUTIVE OVERVIEW — The first major section in the Functional Spec is the Executive Overview. Although it is the first meaty segment in the FS, it is normally the last section to be written, since it gleans its information from the main body of the report. The Executive Overview is geared to senior and middle management and contains, in not more than two or three pages, a brief summary of highlights of the report. Included in the Executive Overview are:
   - The project's scope and objectives, normally a very brief summary of decisions made and project directions decided on at the end of the previous phase.
   - The system overview, which is a thumbnail sketch or brief analysis of the high points of the new system.
   - Any deviations from the project's original resource projections or direction.

The Lasergrafix™ 1200

Now Your DEC Doesn't Have To Think About Printing.

The DEC compatible Lasergrafix 1200 page printer from Quality Micro Systems has a mind of its own. An intelligent controller that takes simple print commands through your normal data stream. And applications firmware to support troff, Tektronix*, Calcomp**, and Versatec***. To take a load off your computer's mind.

Full bit mapped page graphics let you merge text and graphics on the same page. Simplify all your graphics applications—including letter quality word processing. With 12 standard type fonts, and many more available. (You can even create your own.)

All with a crisp resolution of 90,000 dots per square inch, and a whisper-quiet printing speed of twelve pages per minute.

Think about the DEC compatible Lasergrafix 1200. More graphics with less memory overhead. Ask for a demonstration today.

It's the most thoughtful thing you can do for your DEC.

*Tektronix is a registered trademark of Tektronix, Inc.
**Calcomp is a registered trademark of California Computer Products, Inc.
***Versatec is a registered trademark of Versatec, Inc.

QMS®
QUALITY MICRO SYSTEMS, INC.
P.O. Box 81250 Mobile, AL 36689
Telephone (205)633-4300 Telex RCA 26 6013

WHERE IMAGINATION LEADS

☐ Send me the Lasergrafix 1200 Information Packet, samples of mixed fonts, graphics and everything else it does.
☐ Please contact me to arrange a demonstration.

Name: _____ Title: _____

Company: _____

Address: _____

_____

Type of system you are now using: _____

Number of print workstations you are considering: _____

Prime applications for printer: _____

QUALITY MICRO SYSTEMS, INC.
P.O. Box 81250 · Mobile, AL 36689

DT

CIRCLE 0138 ON READER CARD

- Any potential problems, concerns, restraints and constraints should be highlighted.
- A summary of total system impact on corporate resources.
- A summary of the benefits associated with the project (just in case they forgot).
- And in some cases, a copy of the original Project Initiation form.

4. PROJECTED SYSTEM COSTS AND RESOURCE ALLOCATIONS — Most of the projected costs and resources were identified during the first two phases. Unless there was a drastic deviation from those projections, this segment can be optional, or at least provided as a reference segment in the appendices. If this segment is to be part of the main body of the report, include the following:

- A summary of the costs for the first three phases. Show any deviations and be prepared to explain why you are over budget.
- A projection of costs and resources for the next phase. This should be fairly accurate, depending on the changes recommended during this phase.
- A guesstimate of costs and resource projections for the remaining phases.
- Include a GANTT or similar projection chart, if it will help in your presentation.

5. THE DETAILED OVERVIEW — is the guts of the report, written in user friendly language. It should contain everything the user must be aware of to make a decision to determine if the project is still on the right track, within the scope of the original PI request and if it is still feasible and within the realm of corporate resources. It contains:

- An expanded, or more detailed Executive Overview.
- A general narrative, explaining the new project/system in detail.
- The new system workflow, both manual and automated.
- A summary of each affected subsystem (new and old).
- A summary of any new hardware or software requirements.
- A summary of the new system's operating environment.

Although the following segments of this report will contain substantial system detail, this portion should provide at least a brief listing of:

- The inputs to the new system.
- Any special processing functions, calculations or algorithms used.
- The new or redesigned reports or other outputs.
- Any new database fields used or reallocated for new use. This would include any new database(s) to be designed.
- New or revised files to be created or accessed.
- Potential system interfaces.
- Any new or additional communications or teleprocessing programming and equipment requirements.
- Special on-line commands.
- Controls, including auditing procedures.
- And any other items that might prove of interest or reference for the User and/or management.

6. FUNCTIONAL FLOWCHARTS — The inclusion of this segment into the Functional Specifications is debatable. If nothing else, it serves as a checklist or referral to the Project Manager to insure that nothing has been forgotten from the system. Functional flowcharts can be included for the (revised) manual system and new user operational system and detailed to the hardware/software level. In each case, the charts can be detailed to several operational levels:

- The macro or total overview level provides a departmental macro block diagram of the total system.
- The intermediate level decomposes the macro "boxes" into functions within each department or system flow.
- The micro level details the workflow through each intermediate level.

Of course, we can get carried away with our artwork and detailed workflow and carry it down to the micro-micro or micro-micro-micro level, which no one will understand or care to read . . . until an item is missed from one block after implementation!

7. FUNCTIONAL PROCESSING REQUIREMENTS — Smaller systems may include this segment as part of the Detailed Overview Description segment of the specs. However, when designing larger systems, it would be advisable to include a separate segment since the purpose of the Detailed Overview is to define the system without using "bits and bytes" terminology.

The main function of this segment is to detail the functional processing information that makes the system work. Included are the following:

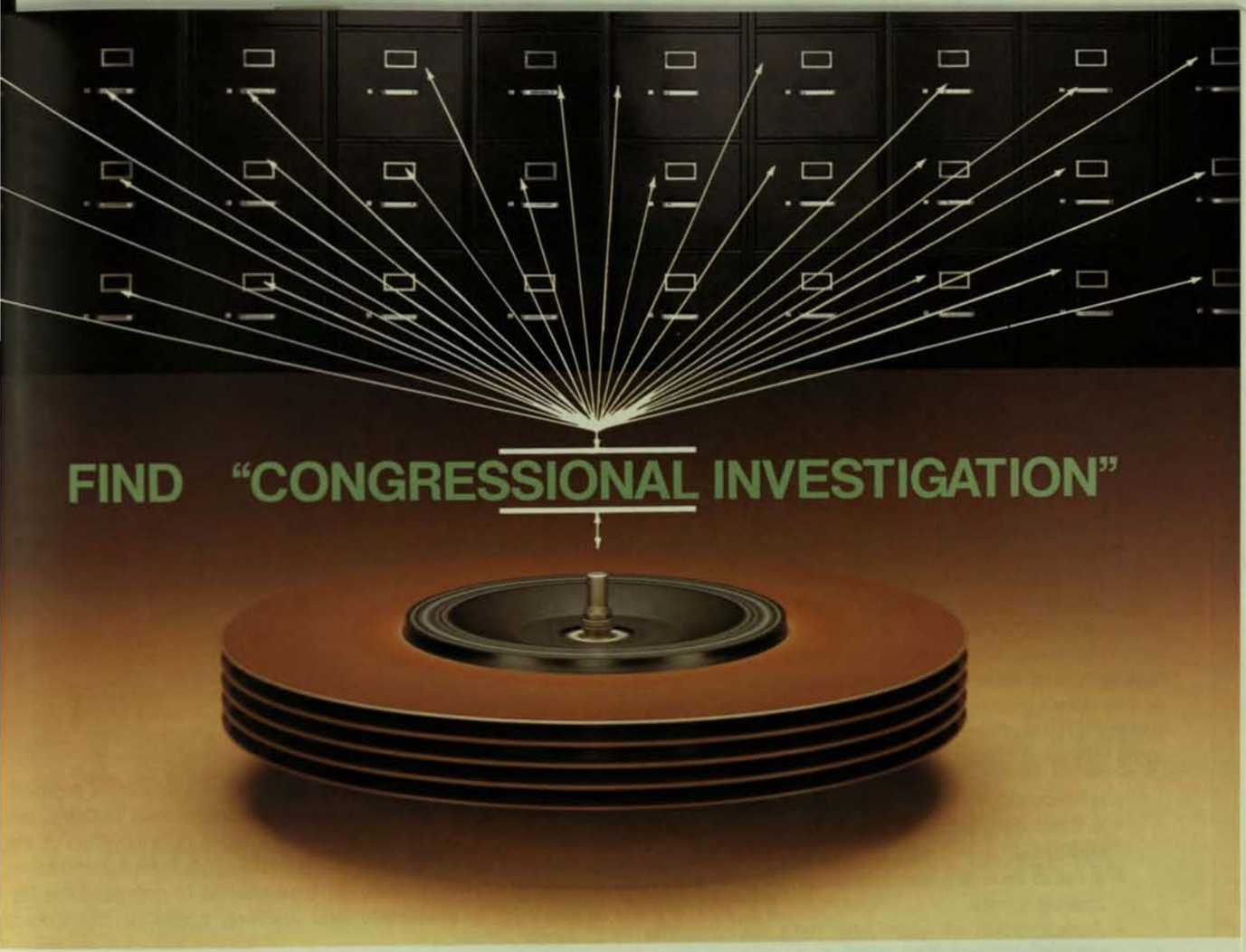
- An expanded functional view of each subsystem specified in the Detailed Overview system narrative segment.
- A detailed look at special calculations and algorithms used to manipulate the system data.
- A detailed list of existing program revisions required.
- A detailed list of interfaces that must be accessed or changed and the impact of those revisions.

8. SYSTEM INPUTS — Can range from a punched card to a remote on-line entry in Timbuktu. A sample layout should be provided on a standard corporate form. If required, the following information should also be included:

- Estimated and projected volumes.
- A brief description of each transaction's usage.
- A transaction (code) grouping, as debits, credits, inquiries, etc.
- A list and definition of each input element. In most cases, the layout form may be enough.
- A cross-reference of each input element to a file, data base, output record or other such usage area. Someone may ask, "What do we need all this for?" If you don't understand why the form has been designed, then how can you justify its existence?

9. SYSTEMS OUTPUTS — Can be classified as anything used by the customer, vendor, user, management, auditors or anyone else directly or indirectly affected by the system. It can be hardcopy or retrievable by an on-line method. It may be paper, tape, magnetic tape/disk, microfilm (fiche) or any other electronic, mechanical or human readable document.

In general, the following information should be sup-



BASIS retrieves information from 1 to 10,000,000 records in nothing flat.

Tired of manually searching through massive files of data? Let BASIS turn your existing files into an automated Information Center.

Imagine a system that retrieves textual information within seconds, the same way a database management system retrieves data. That's BASIS, a proven, interactive retrieval system that does much more than examine titles.

BASIS actually probes to find the key word or phrase you're looking for within any record. Then it displays a list of all records containing the word or phrase you specified. BASIS also displays any or all of the information, and prints hard copy if you want.

BASIS can turn your existing files into an automated Information Center within a few days. *So if you have one or ten million records that need rapid referencing, see what BASIS can do for you.* Call or write for a descriptive brochure today.

Battelle

Software Products Center
505 King Avenue
Columbus, Ohio 43201
Telephone (614) 424-5524

VISIT US AT DEXPO EAST
APRIL 4-6, 1984, BOOTH #828

CIRCLE D175 ON READER CARD

BASIS runs on DEC VAX®, Prime and Wang VS minicomputers, and on IBM, CDC, DEC® and Univac mainframes. DEC and VAX are trademarks of Digital Equipment Corporation.

plied for reference:

- A brief description of the output and its reason for existence.
- A list and definition of each element, cross-referenced to a source similar to the input document.
- The report sequence, frequency, distribution and copies (ply) produced.
- Make certain there is not a built-in redundancy related to several reports, or reports created and sent to people who will never use them.

10. DATABASE/FILE DEFINITIONS — In most cases, this segment will not be of much value to management, except perhaps to those involved in the computer operations area. However, this reference segment will enable the project manager to better understand the complexities of the system and help prevent later additions or revisions by identifying those sometimes "nasty" technical requirements vital to this bits and bytes segment of development.

In general, this segment will include any new magnetic files created or maintained as a by-product of the system. It will specify new database files or fields to be added, accessed, revised or eliminated. It will allow the computer operations section to assess their resource allocations, especially if new hardware is to be added, additional tapes have to be hung, or disk drives manipulated during the course of running the system. This could require additional staff which may have to be hired and trained prior to system implementation.

The following information should be provided:

- A brief description of the new file(s) or database/element(s).
- A list of each new element, cross-referenced to its accessing source.
- A description of each new data element (file or database) to include:
  - -The element name.
  - -A brief element description.
  - -The originating source (input, file or database).
  - -If the element is optional or required.
  - -Specific field characteristics.
  - -Allowable default values.
  - -Editing and validation requirements.

If this seems like more work than should be required during this "early" phase, you may be right. However, it is better to identify any major requirements (or pitfalls) at this stage in the project than during the programming phase when changes or additions can be disastrous!

11. SCREEN (CRT) LAYOUTS — Another overlooked segment of the Functional Spec is the screen layout segment. Most corporations have developed a standardized form to show what the actual screen display will look like. The user should provide major input to this segment since it is his people who will be working with the new system. However, it should be pointed out that the user doesn't always have the best solutions.

It is sometimes advisable to show at least two screen layouts for complex designs . . . just in case there is some point of conflict. In addition, show each access level, that is, if several levels of inquiry are necessary for each screen . . . then show them.

12. ON-LINE COMMANDS, RESPONSES AND PROCEDURES — The purpose of this segment of the report is to group together all new on-line commands, responses and procedures applicable to the user area. This will allow the user to grasp the full impact of the system.

Take, for example, a simple manual system where the line employee's only previous contact with an electronic device was the telephone. With the new system each employee will have a terminal to cope with and a multitude of CRT entries and responses to learn. It would be nice to notify the user of this impending doom so he can beef up his area with the appropriate training, manuals and new personnel as required. You wouldn't want to surprise him on implementation day . . . would you?

Included in this segment are:

- A list of all new on-line commands and any associated access levels.
- A list of all responses.
- The purpose, description and reason for the existence of each command.
- A cross-reference of each command to the data base element accessed or updated.
- A list of error responses and re-entry procedures.
- A list of supervisor or manager (security type) controlled commands and/or responses.
- Any rejection, validation or reconciliation criteria required.

13. CONTROLS — Another often overlooked segment which can prove to be critical are the project controls. The controls listed in this segment are not normally part of the day-to-day production editing and validation procedures. Instead, they are those controls needed to complete the balancing, reconciliation and auditing cycles of the system itself.

Included are:

- Any manual and/or automated balancing, reconciliation, special requirements or audit trails required to complete any normal or special processing cycle(s).
- Any special auditing controls, needs and requirements. This would include header or trailer controls.
- Backup and recovery procedures...including constraints. Provide:
  - -Normal "system down" recovery procedures.
  - -Natural disaster recovery procedures.
  - -Extended power failure recovery procedures.
- List any special security precautions. Include:
  - -System security in general.
  - -Run procedure security.
  - -On-line security.
  - -"Kiting" security and controls.

14. CONVERSION STRATEGY — Conversion strategy should have been identified during the Impact and Feasibility phases. The detail of this strategy should be included in the Functional Specs. Remember, A conversion is a system within a system! Nearly all the same requirements will apply to a conversion as to the total system.

In addition, provide the following detail:

- The method of conversion, that is, parallel, gradual, pilot, hybrid, stand alone, etc.
- Additional staff requirements to convert the data.

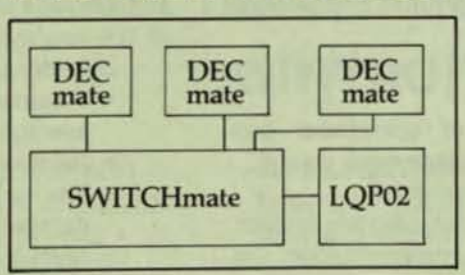
FREE 30-Day Introductory Offer!

The SWITCHmate–$749.00
The SWITCHmate Jr.–$499.00
Dealer Inquiries Invited.

The DECmate Just Got Better.

Introducing The SWITCHmate, Intelligent Printer Switch.

Why fight for the office LQP02 printer when there's the SWITCHmate!™ Intelligent Printer Switch from Gold Key Electronics. Now, you have the ability to form a multi-workstation system with your Digital DECmate family of business computers that can save you time and money. With the SWITCHmate, up to three Digital systems, whether it's the DECmate I,™ or DECmate II,™ or any combination of them, may share one LQP02 printer.



FULLY AUTOMATIC. Using the SWITCHmate is the same as normal printing. The operator simply prints a document in the standard fashion and the SWITCHmate will handle your print request automatically. Each operator may have up to eight documents awaiting the SWITCHmate to print.

DESIGNED FOR THE OFFICE. Because the SWITCHmate is custom designed, it has the features a working office needs:

- Integrated into your Digital system, operators may perform print operations without regard for the current printer activity.
- The SWITCHmate operates unattended. Once installed, no operator attention is required.
- Status lights indicate current activity and those workstations which have a print request pending.
- You may suspend the sharing of the printer and dedicate it to a single DECmate without leaving your workstation.

30-DAY INTRODUCTORY OFFER. Gold Key is now offering a satisfaction guaranteed program for the 3-user SWITCHmate and 2-user SWITCHmate Jr!™ If you are not satisfied, just return the unit within 30-days with no obligation to buy.

SERVICE AND WARRANTY. Service for the SWITCHmate is provided directly by Gold Key. During the 90-day warranty period, from date of purchase, a failing unit will be replaced via Air Express to your door. A toll free number is available 24 hours a day. This warranty service may be extended on a yearly basis at a low cost.

Gold Key ELECTRONICS, INC.

P.O. BOX 186, GOFFSTOWN, N.H. 03045
TELEPHONE: (603) 625-8518

DECmate I and DECmate II are trademarks of Digital Equipment Corporation.
SWITCHmate and SWITCHmate Jr. are trademarks of Gold Key Electronics, Inc.

CIRCLE D183 ON READER CARD

- Any training and educational requirements for existing staff to handle the conversion and post conversion details.
- Give a time frame. If it is to be completed over a period of time, for example region by region, specify what that time frame will be.
- List responsibilities for both user and DP staff.
  -Manuals: writing and editing.
  -Balancing and reconciliation.
  -Post conversion auditing procedures.
- Reconciliation procedures.
- List intra/inter system conversion interfaces required.
- Identify any conversion space needed for the conversion staff and equipment.
- List any required temporary or permanent conversion equipment required.
- In general, provide a detailed set of Functional Specs for the conversion system, as a separate system.

15. IMPLEMENTATION STRATEGY — One can normally assume programming and testing will move along as scheduled, providing the time frames given are realistic and the Functional Specs have been successfully converted and implemented into a super set of Design Specs. With this safely tucked away in our minds, we can define our implementation strategy.

A major factor during this phase is schedule, especially if Federal regulations are of prime concern. But if they are not, and a conversion date has been set, the implementation of the system is quite critical.

A project implementation plan should be supplied to include:

- A projected completion date for total implementation of the system.
- Documentation completion and distribution procedures.
- Completion and distribution of operational procedures for the user and computer areas.
- Training and educational requirements.
- Conversion schedules and their total system impact.

Also, include any additional implementation requirements as:

- Reconciliation and balancing procedures.
- Any special audit requirements and considerations.
- Special procedures and constraints associated with the implementation.
- Any one-time-shot implementation programs required (either temporary or permanent) to successfully implement the system.

Finally, make certain everyone is in agreement with the procedures and aware of areas of responsibility associated with the projected implementation date. It should be emphasized that the date specified for implementation is a guesstimate at this early phase in the project and is dependent on changes or enhancements added to the system. If a Conversion Phase is required, the two phases should be coordinated to prevent a lag in financial data updating, processing and reporting.

16. HARDWARE AND SOFTWARE REQUIREMENTS — During the Feasibility Phase a decision was made by management for one alternative over the others. This decision may have included requirements for new hardware and/or software to be added or used. In order to progress into the Design Phase with confidence, initial contact should have been made with the appropriate vendors to determine certain criteria applicable to the successful completion of the project. Areas of consideration to include, if not already known, are:

- The vendor's name, address and brief history.
- Vendor reliability, based on a personal/corporate survey comparison with other industry users.
- Product reliability, based on the same survey.
- Any delivery constraints specified by the vendor.
- Any contract restrictions or constraints.
- What impact the new vendor product will have on existing hardware or software.
- Will additional training be required? If so, how much, when and by whom.
- Costs, based on alternative vendors, if required. (A future article will concentrate on vendor reliability checkpoints and considerations.)

17. DATA VOLUMES, RETENTION PERIODS AND DISTRIBUTION FACTORS — Some of the information contained in the Functional Specs was based on volumes, retention periods and distribution factors gleaned from the user, management and the industry in general. Since this was a prime consideration in your system design, the information should be reiterated for management review, especially if it was altered or was simply not included before in any document. It will also aid the Operations Department in their planning and scheduling of the data received as a by-product of the new system.

Consider the following:

- Volumes (estimated and projected): peak, low and average.
- Retention, backup and purging requirements.
- The type of media to be produced, its frequency, distribution and copies (ply).
- Special equipment required to store any files or other media associated with the system.
- New equipment location or relocation.

Try to anticipate problem areas. If there is a sore spot, get clarification before continuing the project. It won't go away. It will only perpetuate itself . . . like a cancer.

18. APPENDICES — is a catchall category. Everything that doesn't fit into the main body of the report, or is too extensive to be included or imbedded into the report but is needed for clarification or reference purposes, should be included in this segment.

Provide additional information as:

- Glossaries of specific items or terminology.
- References or cross-references.
- Special codes used.
- Any indexes or cross-indexes that may be of help.
- Letters of reference from the vendor(s).
- Anything else that would be too cumbersome for the main body of the report.
- A table of contents for the appendix, if required.

PUTTING IT ALL TOGETHER

Of course, all this information won't be needed on a two-day project . . . or perhaps even a three-month project. As a matter of fact, some smaller projects may only require one or two phases in their entire life cycle. The point I am trying to get across is the necessity for a standardized format for producing information during the Definition Phase (or any phase) on any project, regardless of its size.

Once management has familiarized themselves with a standardized set of specifications, regardless of the phase, they will expect this type of documentation with every project. As they review each phase end document, they will know exactly where to look to glean the information required to make an intelligent decision as to the status and direction of the project.

Any information not required for a project, whether it is a section of a phase report or several entire phases, can be identified by an omission page or by an omission highlight in the table of contents.

PRESENTING YOUR REPORT TO MANAGEMENT

Just as the playwright must present his completed play to a producer for production, so must the Project Leader present his finished product to management for project continuation and approval.

As the Functional Specs are formulated and put together in segments, they should be reviewed in depth with other team members to insure their continuity flow, validity, and feasibility. Team members should be encouraged to poke holes in the final product before it is reviewed with management. Any changes should be incorporated into the final product. If major holes are discovered, a preliminary meeting may be necessary with the user and/or management to get direction before the presentation.

After the finalized specs are assembled into an acceptable product for distribution, a recap type walk through should be performed, again with team members, to insure nothing has been left out and the product is workable. Only then should the report be distributed to management and a meeting time established, based on corporate and general presentation etiquette. (A future article will cover the fine points and techniques for interviewing, giving a project presentation and communications in general.)

Once your project receives the go-ahead status from management, you can progress into the next phase, the Design Phase.

Editor's Note: Mr. Walsh's unabridged version of System Methodology will appear in book form later this year through Prentice-Hall.

SPAWNING BASIC-PLUS-2 TASKS UNDER RSX-11M

By Tom Bandy, Information Systems Group, Richmond, VA

The BASIC-PLUS-2 CHAIN statement does not spawn the tasks that it "chains" under the RSX-11M operating system. This chain statement uses the RQST$ system directive (BASIC-PLUS-2 RSX/1AS/VMS USER'S GUIDE). Thus, chaining allows only one copy of the task to run at a time. Attempting to chain to a program that is running causes the BP2 trappable error #253 to occur.

There are many ways to avoid having this error "bomb" your program. One method is to trap error #253 and retry until the existing copy of the task is removed from the system. This method may be appropriate for programs that run quickly and then complete, but it is hardly satisfactory for tasks that run for any length of time. (It has been said that no BP2 task runs quickly.)

Another method that allows multiple copies of the tasks requires a simple MACRO program that uses the system directive SPWN$. This method uses the trapping technique described above, but includes an additional step that guarantees the waiting period is always brief.

The task will chain a "spawning" MACRO task that will in turn spawn the desired BP2 task. The BP2 task that chains to the MACRO should trap for error #253 in the event that two users attempt to chain to it simultaneously. If the program sleeps for a second it should successfully chain to the MACRO the second or third try.

The SPWN$ directive requires that the task be installed in the following format: . . . nam. For example, THIRD.TSK as . . . THR. It is also necessary to install the MACRO task so that the BP2 program can chain to it.

The exact SPWN$ call varies slightly between M and M-PLUS systems. The M system requires that MCR be spawned passing the task name through the command line. M-PLUS allows the user to spawn the task directly.

An example of a MACRO to spawn a BP2 task and its requisite assembly in the RSX-11M operating system follows:

```
            .TITLE SPWNER
            .MCALL SPWN$S,EXIT$S
MCR:
            .RAD50 /MCR.../
CMDLIN:
            .ASCII /THR/                ;THIRD.TSK INSTALLED AS ...THR
CHAIN:
            SPWN$S #MCR,,,,,,,,#CMDLIN,#3
            EXIT$S
EXIT:
            .END CHAIN

RUN         $MACRO

MAC>SPWNER,SPWNER=SPWNER

MAC> ^Z

TKB

TKB>SPWNER,SPWNER/-SP=SPWNER

TKB> //

>
```

A REBUTTAL — UNIX REALITIES

. . . continued from page 30

unreadable, your site has some real configuration control problems.

"Reality Number 3: No two UNIX systems look alike." His complaint is that different sites tailor UNIX to their own needs and this causes some divergence. True. That's because UNIX is as easy to modify, to tailor, to customize. "One Size Fits All" fits pretty poorly.

"Reality Number 4: UNIX breaks." True, although not so much as it used to. At any rate, UNIX now includes the tools to fix itself, like "fsck" (File System Check). I don't know of any system that never crashes.

"Reality Number 5: UNIX is user-unfriendly." True. So are CP/M, 360/OS, and everything in between. Operating systems aren't supposed to be friendly, they're supposed to be useful. Application programs are supposed to be friendly.

Now there's a valid complaint: the applications are not friendly. This will change as the market changes from programmers to office workers. Even so, there will never be enough "handholding." WordStar includes 36K of on-line help messages and displays, but people still shell out hundreds of dollars for books, videos and courses on how to use it.

I have personally been involved in installing a menu system on top of several operating systems and have found the UNIX environment to be the easiest to overlay in this manner.

5a) "UNIX manuals are terse to the extreme." True. They are references, not tutorials. There is a great need for better tutorials. Of course, there are also lots of users who never read the manual. If it's long, it's intimidating; if it's short, it's incomplete.

5b) "No two utilities have the same command line convention." True, and one of my personal peeves. So, when I find a useful combination of arguments, I name it and make it an executable image, or put it into my .login file. It's a breeze! VMS is almost as simple, but RSX is a mess by comparison.

"Reality Number 6: UNIX look alikes usually don't." Here, Glover makes a useful distinction between the shell, the utilities, the C language interface, and the assembly level interface. Best advice: try it before you buy it.

UNIX is like Scotch, an acquired taste. I don't like Scotch, Glover doesn't like UNIX. But I don't criticize Scotch for making a lousy screwdriver. ■

PEER NETWORKS OF INTERACTIVE VAX AND IBM COMPUTERS

File Transfer, Mail, and Messages

By Stephen L. Arnold, Joiner Associates Inc., Madison, WI

The system administrator who is not concerned with linking his or her computer to others is becoming an endangered species. The most common type of corporate computer is the big blue mainframe, IBM's System/370 and its successors and imitators. A recurring problem in large and medium-sized companies is, the building of a data highway between departmental VAX computer systems and the corporate mainframe. Possible features of such a link include file transfer, electronic mail, real-time messages and batch job submission. This article looks at methods which are currently available for communicating among VAX computers running the VMS operating system and IBM mainframes using IBM's Virtual Machine/System Product (VM/SP).

IBM VM/SP Is Similar To VMS

Why VM? Simply because VM is IBM's one-and-only interactive operating system for mainframes. In the early 1970s, personnel working in relative isolation at IBM's Cambridge Development Laboratory developed VM/370. This "super operating system" took advantage of the System/370's virtual storage to simulate a number of "virtual" 370s within the software environment of a single real 370. Different operating systems, such as DOS/VS and OS/VS1, could be run simultaneously on the same computer. New applications, and even systems software, did not need to wait until the third shift for testing, but could be run in one virtual machine while production continued in another.

Though there were many advantages, the greatest virtue of VM escaped IBM strategic planners for years. The Conversational Monitor System (CMS), used by systems programmers to maintain VM, was interactive.

The CMS user ran programs by typing their names. These programs could interact with named disk files and the user's terminal, and communicate with other virtual machines by sending and receiving "spool files" via virtual punches, printers, and readers. VM provided the time-sharing function by scheduling the virtual machines to run in rapid succession, giving CMS users the illusion of running their own powerful, dedicated computers.

When moving from VMS to CMS, some new command names must be learned (ERASE is equivalent to DELETE) and the jargon is different (fileid versus filespec), but many concepts and procedures are very similar. During the middle 1970s, while IBM representatives were busy pushing APL and VSPC for interactive use, institutions like Cornell University and the University of Waterloo, Ontario, were giving all their student and faculty users CMS accounts. As the word spread of the superiority of CMS over alternatives like TSO and McGill University's MUSIC, even IBM discovered that VM was its entry into the superminicomputer marketplace. Since VMS was introduced, a host of companies and academic institutions have developed the need to communicate between these two similar operating systems.

Approaches To VM-VMS Networks

But aren't there a host of networking products designed to link minicomputers to IBM systems? Yes and no. IBM's Systems Network Architecture has been so successful in standardizing terminal-host and host-host communication for IBM's MVS and DOS/VSE operating systems that nearly every minicomputer vendor now has SNA software ready or under development. The same can be said for Remote Job Entry (RJE) software and 3270 information display terminal support, emulation and pass-through. IBM has promised a VM compatible with SNA, but because VM was kept in IBM's closet for so long, SNA support in VM will be a long time coming.

Most sites use one of two methods for file transfer between VAX and VM/SP. In the RJE solution, an IBM or compatible machine running MVS and Network Job Entry (NJE) is used as a post office. To send a file, a user prepares a batch job that sends print or punch files to a remote location: e.g., the reader of a CMS user or an RJE station. The batch machine may be a real computer, or MVS can run as a guest operating system in a virtual machine under VM. In either case, an MVS license must be purchased, extra resources are consumed in running the batch job, and the sending user must learn Job Control Language (JCL) to construct the batch job.

A second method is terminal emulation. In this approach, a suitably programmed computer uses a modem to dial the VM system and emulates a live user, uploading and downloading files via the terminal. By providing an additional program on the VM host, error-free file transfer protocols have been implemented, such as KERMIT from Columbia University. The need to learn

the command languages of both computers and the slow speed of terminal communications are the disadvantages of the terminal emulation approach.

The third approach, pioneered at The Pennsylvania State University, is to program a VAX to communicate in VM's native networking protocol. IBM's Remote Spooling Communications Subsystem Networking Program Product (RSCS) is a special operating system that runs in a virtual machine under VM. It transfers files, mail and real-time messages among virtual machines on different real computers like the spool file system transfers files among virtual machines on a single real computer.

In order to attach a VAX-11/780 running VMS to an IBM 4341 running VM at Penn State, Craig R. Watkins developed RSCS software for VMS. (Joiner Associates Inc. recently announced this software product under the jnet trademark.) Watkins' objective was to hook together CMS and VMS by providing DCL commands to originate mail, file transfers and messages, and an easy-to-use method to receive these files, mail, and messages on the VAX. Each VAX was to be a full-routing node in the RSCS network, so routing methods had to be devised. Of course, a number of more mundane matters had to be efficiently handled, such as the translation between IBM's EBCDIC character set and the ASCII characters used on the VAX. To best understand how these tasks are accomplished, a quick review of how RSCS works is needed.

Each VM host in an RSCS network has a nodename, and every virtual machine has a userid, which is unique on

PEER NETWORKS...

its local node. CMS users send files, mail, and real-time messages to other users on their local node by addressing them to the recipient's userid. To communicate with users on other nodes, a CMS user directs the information to the RSCS virtual machine on the local node. The RSCS machine owns the "links," which are binary synchronous communication (BSC) lines from the local node to remote nodes. RSCS constantly tries to read incoming messages and files from its links and virtual reader. If the destination of a message or file is on the local node, RSCS sends the message or spool file to that user. If the destination is a remote node, RSCS consults its routing table, then sends the message or file over the next link to an RSCS virtual machine on the adjacent node. This process repeats until the data reaches its ultimate destination. Electronic mail is a special case of file transfer, and is handled like other files by the RSCS network.

Messages differ from files in that they may contain only one text line, and are immediately retransmitted by each node until they reach their ultimate destination. If the recipient is not logged in or cannot be reached immediately, the sender is informed and must try later. Files, on the other hand, are stored at each intermediate node until the transfer is completed over the subsequent link. This store-and-forward transmission is slower than a COPY or PRINT over DECnet, but is certain to succeed. DECnet requires that all intervening links be up for the duration of the transfer, which can be a rare event in a wide-area, dial-up DECnet.

Commands

To send a message to user Tom on the local VMS system, a jnet user can type:

```
$ send tom "Please submit your
report."
```

To send the same message to a remote node called BOSTON, the command would be:

```
$ send/remote tom boston
"Please submit your report."
```

In either case, if Tom is logged in, the text appears on his screen in seconds. It is immaterial to the sender whether BOSTON is a VAX or an IBM system, and how many intermediate nodes will be traversed to deliver the message. The SEND command of has a superset of the features of the CMS command TELL, and is used on RSCS networks like the VMS PHONE utility on DECnet.

Like messages, files can be sent with no concern for the nature of the receiving system's character set, command language, or file system. To send the FORTRAN program SAMPLE to WENDY at a node called DENVER, a jnet user would type

```
$ send/file sample.for denver
wendy
```

The file would be translated to EBCDIC and sent over the RSCS network with the CLASS, COPIES, DISTRIBUTION, FORM, RECORDSIZE, SPOOLID, FILENAME and FILETYPE all set to their defaults. Alternatively, the sender can explicitly set any or all of these RSCS file attributes. A single SEND/FILE command can send multiple files separately or as a single file, and can even translate tabs to spaces in FORTRAN source files. (IBM's VS FORTRAN compiler cannot handle the tabs that VMS FORTRAN programmers often use.) There are also provisions to send binary files with no translation.

SEND/FILE corresponds to the CMS command SENDFILE.

When inbound files arrive from the RSCS network for VMS jnet users, they are stored in a staging directory, and the recipient is notified by VMS MAIL of the file's arrival and location. The recipient's own directory is never corrupted by files arriving from the network. No user can access files in the staging directory belonging to another user. The staging directory thus serves the same purpose as a CMS user's virtual reader.

VMS users can also use SEND/FILE to submit batch jobs to IBM mainframes. While jnet can directly connect only to VM/SP peer machines, VAX/VMS users can communicate with non-VM IBM hosts attached to an RSCS network, such as an IBM 308x running MVS/NJE. VMS users may prepare MVS batch jobs using EDT, then send them to the IBM mainframe for execution, and direct output over the network to any CMS or VAX/VMS user. Jnet can also send batch output or any other file to any IBM Remote Job Entry (RJE) terminal, IBM mainframe or VAX-11 for printing or punching. Jnet can eliminate the need for 2780/3780/HASP RJE software on the VAX, and uses the same communications hardware usually used for RJE.

Mail is transmitted over the RSCS network using ordinary VMS MAIL commands. Jnet provides a special DECnet object, the mail gateway, which examines all VMS mail sent through a VAX. The gateway examines the address field for an RSCS address, such as GATEWAY::"WENDY@BOSTON". RSCS mail bound for a remote node is forwarded over the RSCS network. VMS mail and RSCS mail for the local node is passed to DECnet for delivery. Mail to IBM nodes is compatible with the CMS RECEIVE command. Mail sent from IBM nodes with the NOTE or PUNCH commands is ultimately delivered to VAX users as VMS mail. The mail gateway can also convert VMS mail from other DECnet nodes to RSCS mail. Thus, the ability to send mail to RSCS nodes is automatically extended from a node running jnet to all the other nodes of a DECnet.

The last major group of RSCS functions handled by jnet is RSCS commands. Users on either VAX or IBM systems can issue commands to any node to learn the system load, the time, the number and names of users logged in, and other network status information. To check the status of links at BOSTON, a user can type

```
$send/command boston query
system links
```

Like RSCS, jnet responds to such commands by returning information to the requesting user via messages.

We make great cost controllers

The New Braegen DEC®-Compatible Disk Controllers: Price/Performance Redefined.

As a DP manager, you must be more concerned than ever with controlling costs. And for those of you with DEC minicomputers, one of the best ways to gain cost control is in the way you control your disk storage. And Braegen's way—with our new family of DEC-compatible controllers—works better.

The BR1630: Better MASSBUS Price/Performance. The BR1630, designed for your VAX 11/780, PDP 11/70, DEC 10 or 20, is the *only* non-DEC MASSBUS controller to emulate DEC's own products. Because there is no imbedding in the CPU—just a simple cable hook-up—installation and down-time are greatly reduced. Because the BR1630 can run mixed-size drives—up to four total—you can customize to trim your costs in hardware, space and field service. And because it features a mapped drive capability that lets the CPU logically address more drives than are physically there, you again get more storage in less space at less cost.

The BR11 and BR750: Better UNIBUS and CMI Bus Price/Performance. Braegen's way works better for a broad range of DEC CPU users. There's the VAX 11/750-compatible BR750. And the PDP 11- and VAX 11/730-compatible BR11. They join the BR1630 in featuring controller microcode of all offered emulations. All have RM03/RM05 software transparency and standard SMD interfaces. All support multiple and mixed drives. And all boast 32-bit ECC and 16-bit CRC.

Braegen Disk Subsystems: Proven Reliability. Our new disk controllers combine better price and better performance. Now, combine these controllers with our removable and fixed disk drives and the resulting 80- to 600-megabyte systems show clearly that Braegen's way works better, in performance, in reliability, in availability and nationwide support.

So call us. The Braegen Corporation, Minicomputer Peripherals Division, 3320 E. La Palma Avenue, Anaheim, CA 92806 (714) 632-9600, Ext. 261. TELEX: 683-406.

BRÆGEN

DEC, MASSBUS, VAX 11, PDP 11, DEC 10, DEC 20, UNIBUS, CMI, RM03, RM05, RP06 are registered trademarks of Digital Equipment Corporation.

CIRCLE D203 ON READER CARD

PEER NETWORKS...

Implementation Details

The implementations of jnet and RSCS are naturally quite different. Much of jnet's work is done by detached processes (daemons) that are started with the system, and which hibernate while waiting for work. Several daemons handle mail routing and local file delivery. There is one daemon to drive each BSC line, and if a local printer is configured in the RSCS network, one daemon delivers files to be printed to the line printer queue. The SEND command works in the users process context, and is responsible for outbound files and messages.

Those jnet programs responsible for routing share a set of routing routines, all of which access the routing database. The database is kept in a global shared segment for efficiency, and contains node and link information, messages in transit, and pointers to files in transit.

All data in transit are in the EBCDIC character set. A reversible translation scheme allows arbitrary binary files to be delivered to VMS users in ASCII, then forwarded to CMS users in EBCDIC with no loss of data. Mail files use standard Internet (ARPA) headers for ease in gatewaying mail to other types of networks.

Jnet works with IBM's RSCS Networking Program Product (Program Number 5748-XP1), Release 2 or later. The VM system is generated as if the VAX were another IBM system running RSCS. No modifications to IBM hardware or software are required.

Any VAX/VMS system with VMS Version 3.0 or later is compatible. Each BSC communications link requires one DUP11 Synchronous Interface, for which jnet provides a driver. Data rates up to 9600 baud are supported.

The distribution kit includes documentation and console media. VMSUPDATE is used to install the software and define the first BSC link and the RSCS nodes downstream of it. The startup file generated by the installation procedure is easily edited to define more complex network topologies. VMS HELP is provided for all forms of the SEND command, and for the network configuration program used by the system manager.

Benefits of RSCS Networking

RSCS has several important advantages over other networking protocols for VM/SP and VMS. Perhaps the most important of these is that CMS and VMS users can exchange information without learning another command language or file system structure. CMS users, in particular, send files, mail and messages to VMS users in exactly the same way as they send them to other CMS users. Sending and receiving information requires no programming or JCL, so managers and clerical personnel can benefit from the mail capabilities of jnet. Programming projects using standard languages, such as FORTRAN 77, can be divided among programmers in both VAX and IBM domains, and project staff members can still freely exchange source and documentation files and mail using jnet and RSCS.

Sites now using the RJE approach to file transfer can upgrade to peer-to-peer communication software while continuing to use the same communications hardware and adding mail and real-time message capability. Those using terminal emulation can forget the complex procedures they currently use and take advantage of error-free transmission at 9600 baud.

With so much attention now focused on microcomputer-mainframe connections, some important issues in minicomputer-mainframe communication were not being addressed. Jnet is an attempt to fill the need for communication between IBM VM/CMS and VAX/VMS users.

Note: The author welcomes your comments at (608) 238-8134.

COMPUTER ROOM FIRE PROTECTION

By Terry C. Shannon, The POISE Company, Inc., Roswell, NM

Although numerous articles have been written about different aspects of computer security and computer room design, the topic of fire prevention and suppression in data processing installations has been neglected. Often this neglect has had catastrophic consequences, such as the total loss of thousands of personnel records in a fire at a St. Louis military computer installation, or the loss of revenue which resulted from the New York Telephone Company computer fire in New York City. This article will address the fire problem and detail the measures that may be taken to minimize the possibility of a fire-related computer room disaster. It will make you aware of some of the fire protection systems commercially available and their relative advantages and disadvantages.

A basic knowledge of fire protection systems and techniques can help you make an informed decision should you be called upon to evaluate a proposed system for your own installation. It may also save you money — by being aware of the capabilities, limitations and usual applications of different fire protection systems, you should be able to select the most cost-effective system for your facility.

Often, little consideration is given to the topic of fire protection beyond meeting the express requirements of insurance companies or the local building code. Fire prevention and suppression measures implemented in a small data processing facility may consist of nothing more than posting "No Smoking" signs in the computer room and hanging a twenty dollar fire extinguisher on the wall. Two reasons help foster this attitude: nobody really expects to have a fire; and, perhaps more significantly, fire protection costs money. As a former system manager, I am well aware of budget constraints and cost justification. And as a former fire protection engineer and firefighter, I am equally aware of the costs of inadequate or nonexistent fire protection measures.

The costs of a major fire, both tangible and intangible, can be very significant. Property damage, equipment loss and the restoration of a building or facility to its pre-fire condition are all costly. In our line of business, the costs of system downtime and file restoration must also be considered. It is impossible to place a dollar value on the intangible costs of injuries, fatalities and the unemployment of workers who have been burned out of a facility. But these costs are very real. A high proportion of businesses which suffer large scale fires close down and never reopen. For these reasons, knowledge of fire protection systems and techniques can be very valuable to you.

The basic forms of available fire protection consist of manually operated fire extinguishers for local application of extinguishant, and built-in automatic fire suppression systems which flood a protected area with an extinguishing agent. This article deals mainly with the second category, but some mention should be made of local application fire extinguishers.

FIRE EXTINGUISHERS

Five types of fire extinguishers, each categorized by its extinguishing agent, are commercially available. They are referred to as carbon dioxide, dry powder, foam, Halon and water charged extinguishers. For data processing applications, you should limit your choices to carbon dioxide and Halon extinguishers. Dry powder is an excellent extinguishing agent, but the powder is

discharged as a stream of very finely divided particles. After using a dry powder extinguisher, you have a major cleanup problem — the powder residue tends to cling to every surface that it touches. You can imagine what this residue could do to all the boards and circuitry in your CPU cabinet. You don't want to even imagine what it could do if sucked into one of your disk drives.

Foam and water extinguishers are not recommended for data processing or electrical applications because of potential damage to equipment, and due to the possibility of electrocution. Both foam and water are excellent conductors of electricity, and it is very likely that the stream of extinguishing agent would contact energized electrical circuits.

Carbon dioxide and Halon extinguishers both discharge their contents in gaseous form, leave no residue, do not conduct electricity, and are very effective. Both extinguishers are good candidates for EDP applications. Carbon dioxide has the advantage of being far less expensive than Halon, but Halon is more than twice as effective on a weight basis as carbon dioxide, and is far less likely to cause thermal shock to computer chips than CO_2. Halon is a superior fire extinguishing agent, but carbon dioxide is a good alternative if cost is your primary consideration.

FIRE PROTECTION SYSTEMS

Unlike local application fire extinguishers, a fire protection system is made up of a group of components that act together to detect a fire, sound an alarm, and discharge an extinguishing agent to suppress the fire. These components include a storage container for the extinguishing agent, a discharge nozzle or distribution piping attached to the container, an automatic detection and release mechanism with manual override, and pressure switches to sound alarms, shut off equipment and close doors and ventilation ducts. Several types of fire protection systems are available for computer room fire protection. Each has its relative advantages and disadvantages, all of which should be weighed before choosing a system best suited to your needs.

Carbon Dioxide Systems

Carbon dioxide is a very effective extinguishing agent for electrical fires and has had widespread use in computer rooms and other facilities with electronic equipment for this reason. It is relatively inexpensive, readily obtainable and leaves no residue to foul chips, memory boards or peripheral equipment. However, a CO_2 fire protection system does have some shortcomings in data processing applications. The primary disadvantage involves personnel safety. When a carbon dioxide system discharges, it floods the area that it is designed to protect with CO_2 gas thereby displacing the normal atmosphere and reducing the oxygen concentration below the 15 percent required to support combustion. A system is usually designed to provide a 30 to 6[illegible] percent concentration of carbon dioxide, which is more than adequate to suffocate people who may be in the area at the time of system discharge. Because the system discharges very

rapidly, the air in the protected area is rendered unbreathable within just a few seconds. For this reason CO2 systems generally have a delay of at least thirty seconds between sensing a fire and discharging. When the system detects a fire, it will sound an alarm to warn people to evacuate the area prior to system activation. A manual override switch is provided to abort system discharge. Some systems are semi-automatic — they employ automatic fire detection and alarm devices but must be manually discharged. Manual discharge eliminates the possibility of personnel being trapped in the protected area when the system floods the area with extinguishant.

While both the automatic time delay and manual system discharge switches are life safety features, they have drawbacks. If a protected area is not occupied twenty four hours per day, a system that must be manually discharged is useless during nonworking hours. And while the built in time delay between alarm activation and agent discharge is critical to personnel safety, it may allow a fire to intensity. Thirty seconds or so may not seem like a long time, but it can mean a great deal in the development of a fire.

Another fact to consider is the possibility of thermal shock to computer chips and boards caused by the extreme low temperature of the carbon dioxide gas. For this reason and the previously mentioned safety factors, I would suggest that total flooding carbon dioxide systems be employed only in areas which are not generally occupied by personnel, such as tape libraries and electrical rooms.

Halon Systems

HALON, which is an acronym for HAlogenated hydrocarbON, is a relative newcomer to the field of fire suppression. Unlike conventional extinguishing agents which act by removing heat, fuel or oxygen from the fire equation, Halon breaks the chemical chain reaction of combustion. This property of halogenated hydrocarbons was discovered early in the century and was put to use as early as 1907 in the form of carbon tetrachloride fire extinguishers. Unfortunately, when carbon tetrachloride is applied to a fire, it gives off highly toxic decomposition products — notably phosgene gas. In the 1930s a number of deaths and serious injuries were attributed to the use of this halogen as an extinguishant, and its use was subsequently outlawed.

The modern Halons, notably Halon 1211 and Halon 1301, do not share this toxicity hazard to any great extent. Developed in the past thirty years, these distant cousins to Freon refrigerant are colorless, odorless and tasteless gases which are effective extinguishants even in very low concentrations. In the concentrations commonly used for fire suppression, the Halons pose no threat to the occupants of protected areas. In its manufactured state, Halon 1301 presents little hazard to individuals exposed to concentrations of 10 percent or less for up to 10 minutes. A three to four percent concentration of Halon is sufficient to extinguish most common fires, and the normal design concentration for a total flooding Halon system is six to seven percent. Thus, it can be seen that Halon is a safe and "user-friendly" extinguishing agent.

The decomposition products of

Halon can be toxic in sufficient concentrations. However, any fire of sufficient size and duration to cause a large percentage of the Halon in a protected area to decompose would undoubtedly result in evacuation of the area before the decomposition products presented a toxicity hazard.

In addition to widespread use in data processing installations, Halon systems are also used in museums and libraries where prevention of damage to rare books and collections is critical. It is also used for such diverse applications as the protection of jet aircraft engines and racing cars. I have witnessed tests of Halon systems and have been in computer rooms when total flooding systems have discharged, and I am convinced that Halon is the best extinguishing agent available for total flooding applications.

The major drawback to a Halon fire suppression system is its expense. Halon costs about five dollars a pound, significantly more than carbon dioxide. However, if you are more concerned with occupant safety and minimal interruption of activity than expense, Halon is the best choice for fire protection.

A typical Halon fire suppression system consists of the same components used in a carbon dioxide total flooding system, with several modifications. Manual discharge of the agent is not normally required, and the time delay between alarm and agent discharge is reduced and in some cases eliminated. Because of the expense of Halon, system actuation is often based on the use of "cross-zoned" detectors. In this scheme, a protected area is divided up into zones, each of which is protected by smoke detectors and ultraviolet flame detectors. When one detector in one zone is activated, an alarm will be given, but the system will not discharge its extinguishing agent. Actual discharge comes only when a second detector in a different zone is activated. In the case of a minor fire, cross-zoning gives personnel time to eliminate the problem and deactivate the system before it floods the protected area with the Halon agent. In the event of a wastebasket fire or the false activation of a single smoke detector, this feature can save a great deal of money, aggravation and disruption.

Automatic Sprinkler Systems

Automatic sprinkler systems have been in use in one form or another for over 100 years. While the concept of automatic sprinklers is somewhat antiquated, the sprinkler system is still the most popular and frequently specified fire suppression measure. Automatic sprinkler systems are far less expensive on a square foot basis than systems employing gaseous extinguishing agents, and have historically been more than 96 percent effective in fire suppression.

There are two basic types of sprinkler systems that are of interest to us. The first and most common variety employs sprinkler heads with fusible metal links that melt when they reach a given temperature and discharge water. This type of system will continue to discharge water until it is shut off by the fire department or the building owner. As computers and electronic equipment are allergic to water, a fusible link sprinkler system is not recommended for such applications.

The second type of sprinkler system utilizes on-off sprinkler heads which are actuated by thermal valves. In this system, the valve on each sprinkler head will open and discharge water when a preset temperature is reached. Unlike fusible link sprinklers, these sprinklers will shut themselves down when the temperature in the protected area falls below the preselected actuation temperature. The main problem associated with sprinkler systems is damage from excess water — not the water which vaporizes into steam and actually extinguishes the fire. The on-off sprinkler system minimizes the amount of water used to suppress a fire and is less likely to cause damage from excessive water. This feature makes on-off sprinklers suitable for data processing installations, particularly very large sites that would be prohibitively expensive to protect with total flooding carbon dioxide or Halon systems.

Combination Systems

In very large installations which have raised floors, a combination fire protection system is often employed. This system will protect the subfloor area where the bulk of the electrical wiring is located with Halon while on-off automatic sprinklers are installed overhead. In such an installation, a fire is most likely to start in the underfloor wiring and cables. The Halon system will deal with such a fire very efficiently and with minimal interruption to data processing activities. Should a fire get out of hand and penetrate the floor or originate above the floor, the on-off sprinklers will quench the fire with minimal damage to the facility and its equipment. In a large installation, a combination system offers effective fire protection at far less than expense than an equivalent total flooding Halon system.

Criteria

When choosing a fire protection system for your installation, several factors should be considered. The most important factor is life safety: your system must first protect people. Data processing equipment is replaceable — human life is not. Choose your system with this thought in mind.

The cost of a fire protection system is a critical factor in system selection. Your goal should be to select a system that will give you maximum desired protection at minimal expense. If you have a choice between an automatic sprinkler system and a total flooding Halon system and are interested mainly in extinguishing a fire, the automatic sprinkler system should give you the protection you need at less cost than a Halon system. If you are also concerned with potential water damage, the additional expense of the Halon system may well be justifiable.

The expense of disaster recovery should also be taken into consideration. While a fire protection system is an expensive item, recovery from a computer room fire can be far more expensive, particularly when you add up the costs of system downtime, cleanup and repair or rebuilding of the computer center, replacement equipment, and the intangible cost of lost business.

Check with your insurance company to determine what steps you must take to protect your computer facility. Your insurance policy may require that you adopt a minimum level of protection. If you do not invoke this minimum degree of protection, a hold-harmless clause in your policy may absolve the insurance company from any

10th ANNIVERSARY SPECIAL OFFER

$995

Our ANSI X3.64 terminal has all the VT100/131 and VT220 features you'll ever need in a conversational terminal. Plus eight colors and vector graphics. Available *now.*

The logical switch to color has never cost less. Now you can get immediate delivery on an Intecolor 2405D single evaluation unit at the special 10th Anniversary price of $995 (U.S. domestic only).

You'll have the advantage of vector graphics on an 80 column by 24 line screen, without sacrificing the most important capabilities you want from DEC's VT100/131 or VT220 terminals. Plus, the 2405D's eight colors convey more information, more quickly and with greater comprehension than VT100/131's monochrome.

ANSI X3.64 compatibility. The 2405D is the ideal replacement or add-on terminal. Highly compatible with VT100/131/220 and numerous other ANSI X3.64 terminals, the 2405D is easily integrated into any ANSI X3.64 environment. It includes ASCII codes and a VT52 mode. The 2405D is also compatible with Intecolor's 2427 bit-mapped color graphics terminal.

With all the features you need. Terminal based vector graphics. Data transmission baud rates from 50 to 19,200. English language menu set-up mode. Non-volatile set-up memory. *Two* full pages of screen RAM. In-line CRT. Auto degaussing. Powerful, 6MHz 8085 CPU with four hardware interrupts. Detached keyboard. Our 10th Anniversary $995 offer includes 12 function keys (with 36 programmable functions), normally optional. Another 12 keys are optionally available for $100. Shipment less than 30 days after acceptance of order.

Plus Intecolor Quality. Since 1973, Intecolor has been a pioneer in high quality color graphics terminals for industrial/scientific markets. And we are continually setting new standards of reliability.

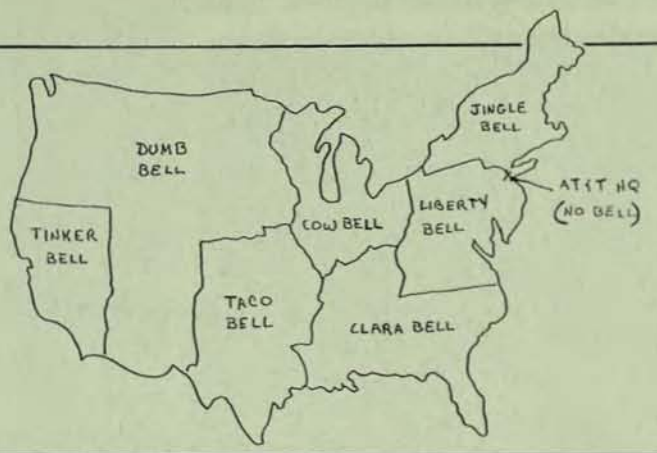
10th Anniversary price good through 1/31/84. After that, our regular single-piece price of $1295 (100 piece: $1075) goes into effect, and the first 12 function keys are $95 with a total cost of $1390. Save $395 by acting now! Take full advantage of the color, vector graphics and ANSI X3.64 compatibility of the Intecolor 2405D at this special price.

For the name of the distributor or sales representative in your area, or for complete specs, ask about our 10th Anniversary $995 special: **Call 1-800-241-7595.**

Intecolor

AN INTELLIGENT SYSTEMS COMPANY

Intecolor Drive, 225 Technology Park,
Norcross, GA 30092, TWX 810-766-1581



AT&T Announces Names for New Regional Holding Companies . . .

liability resulting from a fire. If you lease your EDP equipment, read the terms of the lease very carefully. Most lessors require that you protect your leased equipment even though you do not own it.

CONCLUSION

Some of the fundamentals of fire protection systems have been covered here, but this is not an in-depth study. Should you be considering the purchase of a fire protection system, your best sources of further information are fire protection system vendors, major fire protection equipment manufacturers and independent fire protection engineers. Your local fire department and building code commission as well as your insurance carrier may also be of assistance. It is hoped that the information in this article will make you more conscious of the importance of fire protection and suppression in a data processing environment, and provide you with some ideas for increasing the level of fire protection and safety at your installation.

Mr. Shannon holds a degree in Fire Protection Technology and has worked as a fire protection system designer for consulting engineering firms. He has also served as a fire investigator for the Onondaga County Arson Task Force and as a line officer in the Jamesville Volunteer Fire Department in New York State. He now works as a technical writer and systems specialist for the POISE Company, Inc. in Roswell, New Mexico, a leading producer of academic software for colleges and universities that utilize DEC PDP and VAX computer systems.

X11MNT

By Terry Ridgers, Systems Analyst
St. Clair River Works
Corunna, Ontario

Here is a program (X11MNT) written in BASIC +2 to log a user into any account without a password. X11MNT is a nebulous name used to confuse its true purpose. To run X11MNT the user responds with a device, an account number and any file name and extention (i.e., SY: (1,92) PASS.WRD). If a non-existent account is entered the user is logged out.

We use X11MNT extensively to log batch control commands into privileged accounts. The following scenario generally takes place:

1) The user logs into a non-privileged account; say (33,2).

2) Through menu control the user runs a BASIC +2 program; say (1,92) DTRLST.TSK <232>.

3) The user wants a DATATRIEVE report run by various criteria (selected through DTRLST.TSK).

```
Cross-Reference Listing of X11MNT on ??-Mar-83 at 11:31                    Page 1

10  ON ERROR GOTO 25000                        !TRAP GENERAL ERRORS            &
    OPEN "KB:" FOR INPUT AS FILE 12%           !OPEN KEYBOARD FOR INPUT        &
    T.1$ = NOECHO(12%)                         !DISABLE KEYBOARD ECHO          &
    INPUT LINE #12%,ACCOUNT$                   !ENTER LOGIN ACCOUNT AS:        &
                                               !   DEV:[n,n]PASS.WRD :"        &
    CLOSE 12%                                  !CLOSE KEYBOARD OPEN            &
    PRJ.ACC$ = SEG$(ACCOUNT$,POS(ACCOUNT$,"[",1%),POS(ACCOUNT$,"]",1%))        &
    PRJ%=VAL(SEG$(PRJ.ACC$,POS(PRJ.ACC$,"[",1%)+1%,POS(PRJ.ACC$,",",1%)-1%))   &
    ACC%=VAL(SEG$(PRJ.ACC$,POS(PRJ.ACC$,",",1%)+1%,POS(PRJ.ACC$,"]",1%)-1%))

20  T.1$ = SYS(CHR$(6%)                      + !OBTAIN [?,?]PASSWORD           &
            CHR$(14%)                        + !CODE TO READ ACCOUNTING DATA   &
            CHR$(0%)                         + !READ ACCOUNT SPECIFIED BELOW   &
            CHR$(SWAP%(0%))                  +                                 &
            CHR$(0%)                         + !READ ACCOUNT INFO ONLY         &
            CHR$(SWAP%(0%))                  +                                 &
            CHR$(ACC%) + CHR$(PRJ%)          + !FULL ACCOUNT NUMBER            &
            CHR$(1%)                         + !DON'T RETURN # OF BLOCK DATA   &
            STRING$(13%,0%)                  + !RESERVED (SHOULD BE ZERO)      &
            SEG$(FSS$(ACCOUNT$,1%),23%,26%)    +                               &
                                               !DT: AND UNIT # AND UNIT # FLAG &
            STRING$(4%,0%))                    !RESERVED (MUST BE ZERO)        &
    SY.PSW$ = SEG$(T.1$,9%,12%)                !SAVE PASSWORD IN RAD50 FORMAT  &
    T.1$ = SYS(CHR$(6%) + CHR$(5%))            !LOGOUT                         &
    T.1$ = SYS(CHR$(6%)                      + !LOGIN TO ACCOUNT [?,?]         &
            CHR$(4%)                         + !LOGIN CODE                     &
            STRING$(2%,0%)                   + !RESERVED (SHOULD BE ZERO)      &
            SEG$(FSS$(ACCOUNT$,1%),5%,6%)      +                               &
                                               !ACCOUNT NUMBER                 &
            SY.PSW$                          + !PASSWORD                       &
            STRING$(20%,0%))                   !RESERVED (MUST BE ZERO)        &
    GOTO 32767

25000 PRINT "ERROR #";ERR;" ";ERT$(ERR);" AT LINE: ";ERL                       &
      RESUME 32767

32767 CLOSE 12%                                !CLOSE KEYBOARD                 &
      END                                      !END OF PROGRAM
```

4) DTRLST creates and queues (1,92) DTRLST.CTL <40> to batch to run DATATRIEVE with the selected criteria.

5) Because DTRLST.CTL will log into (33,2), X11MNT is used to log the Batch Job Control program into (1,92) and then DATATRIEVE is executed in Batch.

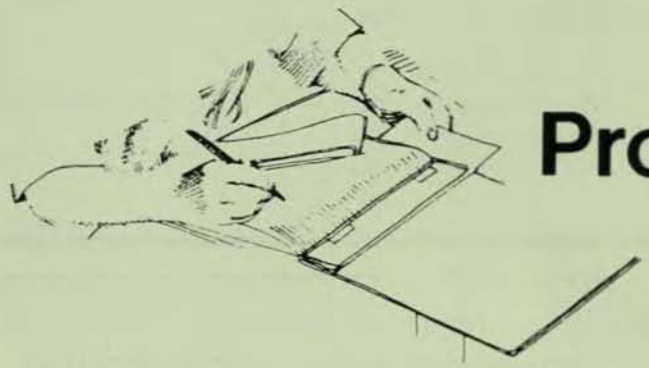
6) The following commands are executed in DTRLST.CTL:

.

.

.

$DELETE (1,92) DTRLST.CTL
$EOJ



Programmer's Notebook

PLANNING FOR A SUCCESSFUL AUTOMATION PROJECT

By Richard Lowe, Software Techniques, Inc.

Mesa Consolidated Water District serves 140,000 customers in the thriving city of Costa Mesa, near Newport Beach, California. Within the water industry, an industry traditionally slow to innovate and tangled in regulations, Mesa has always been somewhat of a leader. They were the first in the Southern California Basin to install an ozone treatment facility, allowing them to convert low-grade ground water to high-quality drinking water. In the areas of public information, and legislation, Mesa's programs are second to none.

So it was no surprise when, in 1975, Mesa put an end to a decade of mechanical billing. Although these electro-mechanical dinosaurs (made by Burroughs and NCR) performed adequately, they were not very flexible. The reports Mesa needed in order to manage their district effectively had to be compiled manually. The district was growing, and customer billing began to take longer and longer. Mesa would have to automate, or drown in a pool of paper.

In 1976, Mesa replaced their semi-automatic billing equipment with computer technology; they went online with the Xerox service bureau. But, after using Xerox for only one year, they were still dissatisfied. The batch-mode system was inflexible, the costs were high, and Mesa was still compiling their management reports by hand.

Mesa's First Computer

In 1978, Mesa installed their own computer. They contacted a local Digital OEM who, after a brief analysis, sold them a PDP-11/34 with two RM02 disk drives, seven VT52 terminals, a line printer, and RSTS/E.

The OEM developed custom application software for Mesa, written in BASIC-PLUS with DMS500 records management. The software was delivered without documentation of any kind, but Mesa was assured that it would follow.

Because the system seemed to meet their needs, Mesa switched over to the new system and began running customer bills. But, after a few months, mysterious "out of balance" problems arose and strange bugs appeared.

By early 1980, Mesa recognized that their system still was not powerful enough. Although they had plenty of computer hardware, the system was difficult to use, data would frequently get lost, and they couldn't find information they needed when they needed it. Attempts to fix these problems caused other problems, and Mesa's satisfaction with the system began to spiral downward.

Finally, two years after installation, a disk crash put an end to the system. The backup procedures were not adequate to prevent the loss of huge amounts of data, including the current versions of the source programs.

A Smarter Approach

Mesa stepped back, took a deep breath, and tackled the problem again. Smarting, but smarter, Mesa realized that in order to be an asset to their business the ideal system must:

1. Reduce Billing and Accounting Errors,
2. Improve Customer Service,
3. Hold-the-Line Against Cost Increases,
4. Effectively Manage a Growing District.

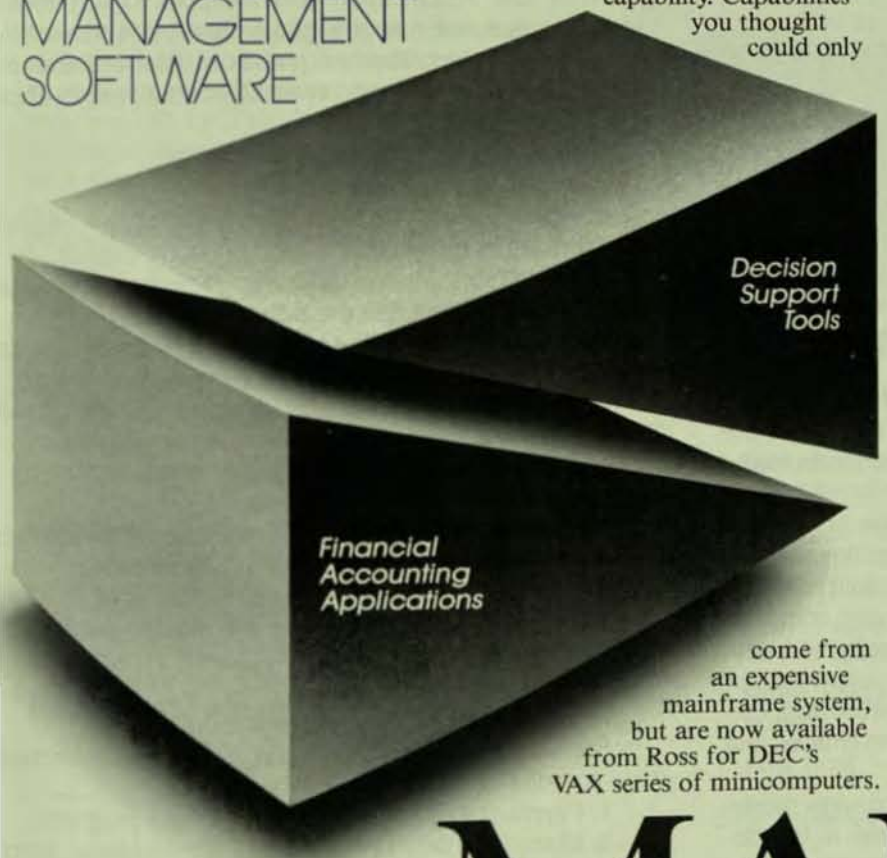
And so, with these goals clearly in mind, Mesa embarked on a two-year program which changed the face of utility billing. The result was a combination of data processing and office automation tools designed specifically to meet the needs of water companies. They called this system WBS-11.

Turn Your VAX into a Total FINANCIAL Solution

INTEGRATED FINANCIAL MANAGEMENT SOFTWARE

ROSS INTEGRATES KEY FINANCIAL FUNCTIONS

MAPS is a new generation of software merging decision support tools and financial accounting applications into a powerful, flexible financial management capability. Capabilities you thought could only come from an expensive mainframe system, but are now available from Ross for DEC's VAX series of minicomputers.



And all Ross products are fully on-line, interactive and bring you multi-user access. What's more, we let you link microcomputers with minis, using the latest in distributed technology to tailor computing to your specific needs.

- MAPS/MODEL—Powerful financial modeling
- MAPS/GRAPH—Quality business graphics
- MAPS/DB—Flexible data base management
- MAPS/Pro—Micro modeling and graphics
- MAPS/GL—Comprehensive general ledger
- MAPS/AP—Full featured accounts payable

OUR SERVICE AND SUPPORT MAKE THE DIFFERENCE

We bring a commitment to our customers and products that reflects our 12 years of consulting experience. Our staff of experienced professionals can work closely with you to install our products, and to provide the ongoing training and support that make them true financial solutions.

And Ross offers a choice on how you use MAPS. You can license MAPS or use it through ROSS/NET, our worldwide, 24-hour remote computing service.

MAPS from Ross Systems. The single source for integrated financial management software.

DEC and VAX are trademarks of Digital Equipment Corporation. MAPS and ROSS/NET are trademarks of Ross Systems, Inc.

MAPS/GL was developed by Price Waterhouse under the name "FM80" and adapted for the VAX by Ross Systems, Inc.

MAPS™ from Ross Systems

Financial Software Solutions

ROSS SYSTEMS

DISTRICT OFFICES:
Palo Alto, CA
San Francisco, CA
Los Angeles, CA
Dallas, TX
New York, NY

CORPORATE HEADQUARTERS
1860 Embarcadero Road
Palo Alto, CA 94303
(415) 856-1100

CIRCLE D167 ON READER CARD

Programmer's Notebook

In mid-1981, Mesa asked Software Techniques, Inc. (STI) to work with them to develop WBS-11. They wanted the design for WBS-11 to be based on their old system because, although this system suffered from a number of problems, it provided the basic functions required by their district.

The problem was complex:

- Billing and accounts receivable information must be maintained on more than 140,000 customers. Bills must be sent to each of these customers on a bi-monthly basis. If the customer account is delinquent, a special bill must be mailed; eventually, the water must be shut off.
- Each customer can have one or more meters (which measure the amount of water used) on their property. The readings from these meters must be manually or automatically read and entered into the automated system to determine the amount to be billed to the customer. In addition, the meters must be tested periodically, and repaired or replaced if necessary.
- A backflow device is attached to each meter. This device prevents used water from flowing back into the water system. These devices must be tested periodically, and repaired or replaced if necessary.
- The system was designed to combine online and batch technology. From 1 to 99 meter readings (known as a batch) could be entered at a time. Each batch could be listed, adjusted and modified at will, giving each user complete control over the data before it was permanently added to the system. The batch was updated to the system master files only after the user was sure the information was correct. After the weekly meter readings had been entered and updated, the bills were printed.
- Two years of meter readings and accounts receivable history must be maintained for each meter in the district. This history is used for forecasting and to aid in tracking down complaints.

Analysis

Our first task was to determine Mesa's automation goals. When we determined what they hoped to solve with this new system, it became clear that most of the problems revolved around the efficiency and accuracy of the old system.

We asked the users to describe each function of the system in simple terms. They were asked what tasks they perform, where the information to do the tasks comes from, what action results from these tasks, and so on. From this, a basic flowchart of the steps required to perform the functions was developed.

Each piece of information was analyzed: who needs it, what is it used for, where it comes from, what is done with it, etc. The information gained from this analysis was compiled into a specification describing how the system appears to the user. All functions, reports, screens and data bases were clearly defined in this specification.

When the specification was completed, it was presented to Mesa, where each of the users reviewed it with his or her manager. Any problems that were discovered were reported in regular review meetings, where any disagreements or concerns could be freely voiced. Following each meeting, modifications were made to the spec and it was reviewed again. This process was repeated until all parties were in complete agreement.

Coding

After the specification was accepted, coding began. This was scheduled in several phases to provide greater control over the project. The first phase consisted of the simple maintenance programs and their associated reports. These included:

1. Customer Maintenance and Register.
2. Meter Maintenance and Register.
3. Miscellaneous Charges Maintenance and Register (special charges to be billed to particular customers).
4. Rate Table Maintenance and Register (the calculations upon which to base the billing amount).
5. Parameter Maintenance (used to enter information about the district using the WBS-11 system).

In the next phase, the billing functions were coded. These included:

1. Meter Reading Entry program.
2. Meter Reading Register.
3. Skipped Accounts List (which lists all of the accounts not billed within a particular area and time period).
4. Meter Reading Update (which updates all meter information entered in a batch into the master files).
5. Billing Register.
6. Bill Printer.
7. Delinquent Bill Register and Printer.

Following the billing functions, the payment functions were coded. This phase consisted of:

1. Payment Entry, Register and Update programs.
2. Miscellaneous Credit/Debit Adjustments Entry, Register and Update programs.
3. Deposits Entry, Register and Update programs.

After the payment functions were completed, miscellaneous functions were coded. These were:

1. Closed-Account Entry and Register programs.
2. Deposit Refund Entry, Register and Update programs.
3. Closings Update.

Special inquires and functions were coded last. These functions include:

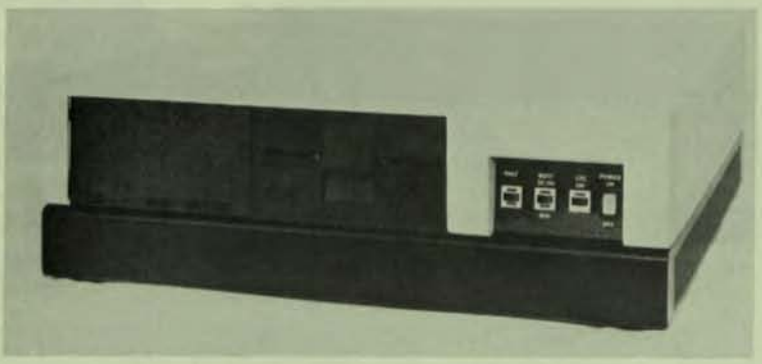
1. Customer Account Locator (allow a customer record to be looked up by customer number, meter number, customer name "Sounds like" code, and street address).
2. Account History Inquiry (display up to the last two years of accounts receivable history for a customer).

LSI-11 USERS:

SIGMA OFFERS
COMPLETE SYSTEMS ENCLOSURE

SA-H110

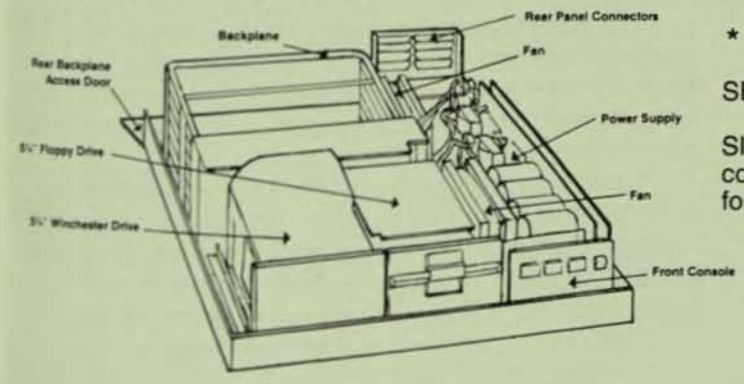
The SA-H110 is a styled and modern tabletop cabinet which provides the systems integrator an attractive and affordable means for building a complete LSI-11 Q bus* or Multibus* system. It includes a heavy duty switching power supply, a full 8-slot, quad Q bus or 8-slot serial/parallel priority Multibus backplane, an operator's control console, two fans for forced air cooling, and mounting and power for two full-sized 5¼" drives.



The Q bus backplane, which supports all LSI-11 type processors, includes 22-bit addressing and full Q bus termination allowing use of all compatible memories, communications interfaces, device controllers and multifunction boards.

A connector panel at the rear of the chassis provides mounting for eight RS-232 connectors for convenient egress of I/O cables. Additional connector mounting on the rear door is optionally available.

SA-H110 Configuration



FEATURES:

* Complete system can be contained in one compact chassis.
* Includes either quad, 8-row Q bus backplane or 8-slot serial/parallel priority Multibus backplane.
* Supports most 5¼" full height or slimline drives including Winchester disk drives, removable cartridge drives, floppy disk drives or cartridge tape transports.
* Power supply provides 40 amps of 5V, 10 amps of 12V in addition to all power fail and control signals.
* Front operator panel provides convenient operator control.
* Low Cost.

SEE SIGMA FOR YOUR ENCLOSURE NEEDS!

SIGMA offers a complete line of DEC* compatible disk controllers, interface devices and system enclosures for the LSI-11.

Sigma Information Systems

6505C Serrano Avenue, Anaheim CA 92807
Telex 298607 SGMA (714) 974-0166

CALL YOUR nearest SIGMA SALES OFFICE for information on all our products.

Orange CA (714) 633-0652 / 771-7320 • San Jose CA (408) 985-9112
Arvada CO (303) 431-0551 • Orlando FL (305) 628-8896
Boston MA (617) 938-1310 • Minneapolis MN (612) 881-5015
Albuquerque NM (505) 294-5423 • Beaverton OR (503) 644-0304
Houston TX (713) 451-0466 • Vienna VA (703) 448-6800
Bellevue WA (206) 454-4600 •Quebec Canada (514) 331-5980

* Multibus is a registered trademark of Intel Corporation and Q bus and DEC are registered trademarks of Digital Equipment Corporation.

CIRCLE D121 ON READER CARD

Programmer's Notebook

3. Account Usage Inquiry (display up to the last two years of meter readings for a customer).

4. Master File Purge (physically delete records which have been marked for deletion from any master file).

5. History Unload (copy old history records from master files into temporary files and delete the old history record from the master files).

6. Special Magtape Backup functions (as transactions are entered into the WBS-11 system, they are backed-up onto magtape. These transactions can be restored later if the system suffers a catastrophic failure).

During the final phases of coding, the documentation was completed. Three documents were delivered with the system:

1. User's Guide

This document tells the user how to operate all functions of the system except those functions reserved for use by the system manager. Each step required to perform each function is discussed in the order it must be performed.

2. Manager's Guide

This document describes the data structures required by the system, the menu system and any functions reserved to the system manager.

3. Technical Manual

This document describes how to build the WBS-11 system and the design philosophy. It is designed for use by a programmer who needs to add new functions to the system.

Training

The initial training sessions were designed to dispel the myths about computers — what they can and cannot do. We described the various components of the computer system and how they interact, giving the users a better feel for the tool they were being asked to use.

Later, the relationships between the various data structures were discussed so that the users understood exactly what occurred before, during and after something was processed. Both the automated functions and the manual procedures necessary to support them were described fully.

Last, a few of the key users were taught to use the more sophisticated tools (like DATATRIEVE).

Conclusions

As we have repeatedly discovered, the key to a successful automation project is planning. Due to the large amount of planning that went into the WBS-11 project, we were able to set Mesa's expectations to a reasonable level, and then to meet those expectations.

OVER 500 SOFTWARE COMPANIES ARE AT

SOFTCON

TUESDAY–THURSDAY, FEBRUARY 21–23, 1984, SUPERDOME, NEW ORLEANS

If you buy or specify a lot of software for your company, government agency, or educational institution, attending Softcon—the international tradefair and conference for the software industry in New Orleans—will be the most productive and profitable trip you'll make this year.

Nearly 600 different software companies using 1000 displays have filled the massive (400,000 square foot) Louisiana Superdome in New Orleans to showcase over 12,000 software products. These companies will use Softcon to unveil some of the most revolutionary new software products in the industry's short history. More than 70 seminars, panel discussions, forums and workshops, presented by the world's most renowned computer authorities will update you on the newest products and industry trends.

You can learn more in one or two days from the experts at Softcon than you could in months by reading trade journals or listening to salesmen. You'll see, try out, and compare brand new software which expands and increases the user-bility and productivity of your existing hardware, and you'll meet new vendors.

Registration is $30 for exhibits only and $195 for admission to exhibits and the conference program. For complete information about attending or exhibiting, call or write Northeast Expositions, 822 Boylston Street, Chestnut Hill, Mass. 02167. Tel: 617-739-2000 or 800-841-7000. Because it's carnival season, hotels are filling fast, so call today.

SOFTCON EXHIBITORS

AS OF DECEMBER 1, 1983, MORE ARE BEING CONFIRMED DAILY.

A.J.&M. Associates, ADS Software, AHA, Aardvark/McGraw, Accent Software, Access Technology, Accountants Microsystems, Adapt, Addison Wesley Publishing Co., Advanced Data Institute of America, Advanced Legal Software, Advanced Technology, Adventure International, Agdata, Agri-Data, Allenbach Industries, Alpha Software, Alpine Datasystems, American Business List, American Business Systems, Anderson-Bell, Anitek, Ann Arbor Software, Apple Computer, Applied Data Communications, Applied Data Research, Articulate Publications, Artificial Intelligence, Artsci, ASAP Ashton-Tate, ASI, Attache Software, Autodesk, Avant Garde, Axel Johnson/Stenmos Ltd. · BASF, Baker Hill & Company, Balcones, Baudville, Beaman Porter, Bellesoft, Berge Software, Best Programs, Bluebird Systems, Boeing Computer Services, Boston Documentation Design, Boston Educational Computing, BPI Systems, Brag Systems, Bristol Information Systems, Brown Disc Manufacturing, William C. Brown Publishing Company, Bruce & James Program Publishers, Burroughs Corporation, Lloyd Bush & Associates, Byte Publications · Cal Micro, CDEX, CES Publishing, CV Mosby, Cactus Software, Call Manager, Cap-Gemini, Capro, Cimarron Div./Standun Control, Coade, Commodore Business Machines, Communications Packaging, Communications Research Group, Comprehensive Software, Compute Magazine, Computech Group, Computer Advanced Ideas, Computer Company of America, Computer Development, Computer Distributors, Computer Intelligence, Computer Marketing, Computer & Software Education, Computer Software News, Computer Technology Center, Computer Technology Group, Computerized Business Systems, Computing Canada, Concentric Data Systems, Condor Computer Corporation, Construction Data Control, Continental Software, Coopers & Lybrand, Counselor Systems, Courseware, Creative Freedom Software, Cube Computer Corporation, Cullinet, Cyma · DLM Teaching Resources, DVC, Dagar, Data Base Research Group, Data Management Assoc., Datamost, Datamension, DataPro, Data Systems, Dayton, Digital Equipment, Dilithium Press, Disco-Tech, Dow Jones, Dr. Dobbs Journal, Dynaquest (Hill & Knowlton), Dynatech, Dysan Corporation · EWDP Software, Eastman Publishing, Electronic Protection Devices, Emerging Technology, Enigma Logic, Eric Software, Excalibur Technologies, Execucom, Executaware · Fastware, Ferox Microsystems, Financier, First Software, Fisher Business Systems, Flexware, Fliptrack Learning Systems, Forethought, Formaster, Forth, Fox & Geller, Fox Research, Fuji Photo Film, Funk Software · GAI Systems, GMS Systems, General Datacom, General Looseleaf, Geocomp, Gessler Publishing, Gilchrist Software, Gnosis, Government of Ontario, Great Plains Software, Gryphon Micro Products · Handle Corporation, Hansen Management Systems, Harper & Row, Harvard Software, Hawkeye Grafix, Hayden Software, Hayes Microcomputer, Henco, Heritage Holdings, Hermenet, Hewlett Packard, Hiller Industries, Houghton Mifflin Company, Hughes Aircraft, Human Edge Software Company, Human Engineered Software · IBM Corporation, I.B. Magazette, IEEE Computer Society, IMSI, ISO World, IXI Sales & Services, Ideaware, Individual Software, Industrial, Infocom, Infodata, Info Design, Information Builders, Information Unlimited Software, Infosound, Infotran, Ingram Book Company, Insoft, Intelligent Designs, Interactive Tele-Marketing, Intermedia, International Micro Systems, International Software Systems, Ivy Hill Corporation · Jones Colad, Kangaroo Software, KPB Industries, Kent Marsh Ltd., Key Enterprises, Key Software, Knoware, Koala Technologies · Laboratory Microsystems, Lantech Systems, Lassen Software, Leading Edge, Learning Well, Lewis Lee Company, Lifetree Software, Lightning Software, Living Videotext, Lone Star Video, Lotus Development · MAI/Basic Four, MBP Software & Systems Technology, M&C Systems, MCBA, M2C5 Systems, MIS Week, MMCA (Mini Micro Computer Application), M&T, MacMillan Publishing, Management Decisions, Mathtech, McGraw Hill, Media Systems Technology, Medicomp of Virginia, Memron, Menlo Corporation, Meusoft, Micro Associates, Micro Business Applications, Micro Courseware, Micrografx, Micro Information Publishing, Microline, Micro Manufacturing Systems, Micro Market World, Micro Software International, Micro Pro, Micro Rim, Micro Strategies, Microstuff, Micro Tax, Millenium Group, Milton Bradley, Mini Micro Systems, Minuteman Software, Mirror Images, Mosiac Software, Mossland Technology, Mountain Computer, Multisoft, Multisolutions, Mycroft · NCR Corporation, Newline, Nitel, North American Business Systems, Norton Software, Noumenon Corporation · OA Software, Odesta, Optic Graphics, Omega Microware, Omnisoft, On Line, Open Systems, Oracle, Osborne-McGraw, Ovation Technologies · PCM, P.C. Telemart, P.C. World, Pacific Data Systems, Palantir Software, Parameter Drive Software (PDS), Paramis, Peachtree Software, Penquin Software, Perfect Software, Personal Cad Systems, Phaser, Phillips Publishing, Plantronics, Prentice Hall, Primavera Systems, Prime Computer, Professional Conference Assoc., Professional Software, Psychotechnics, Psycomp, Publisher's Weekly (R.R. Bowker), Publishing Technology · QED Information Sciences, Quadratron, Quarterdeck Office Systems, Quest Research, Queue, Quicksilva · R.R. Bowker (Publisher's Weekly), R.R. Donnelly, Reader's Digest Microcomputer, Real Estate Application, Real World (formerly MBSI), Red Wing Business Systems, Reference Magazine, Relational Data Base Systems, Reliance Plastics & Packaging, REMS Publishing, Reston Publishing, Rocky Mountain Software Systems, Ryan McFarland Corporation · STSC, SMC Software Systems, SAMNA, Howard Sams Company, Satellite Software International, Scarborough Systems, Schuchardt Software Systems, Scott Foreman, Select Information Systems, Sentinel Computer Products, Sight & Sound, Signature Software, Silicom, Simplicity Software, Small Business Computers, Sofstar, SoftCraft, Soft Kat, Soft Letter, Softech Microsystems, Softpac/Univex International, Softsel Computer Products, Soft Set, Softsmith, Softrend, Softron, Software Arts, Software Automation, Softwarebanc Dealer Services, Software Digest, Software Distribution Services, The Software Group, Software Ireland, Software Libraries, Software Models, Software News, Software Publishing, Software Solutions, Software Strategies, Software Supermarket, Software Ventures, Software Wholesalers, Softword, Sony Corporation of America, Sorcim, The Source, State of the Art, State of the Art Systems, Sterling Swift Publishing, Summit Software, Supersoft, Sybex, Symantec, Symmetric Software, Systems Automation, Systems Management Assoc., System Solutions · T-Maker, TCS Software, Tab Books, Taurus Software, Techland Systems, Technique Learning, Telesun, Teletech, Teleware, Texor Corporation, Think Technologies, Thoughtware Institute, 3M Data Recording Products, Timberline, Timebase, TOM Software, Trac Line Software, Tri Micro International, Tronix, Trutec Software · Unicomm, Universal Custom Products, Universal Software Associates, Uveon · Value Line, Vault Corporation, Verbatim, Visicorp · Wadsworth, Wadsworth Electronic Publishing, Webb Company, Weekly Reader Family Software, Western Electric, Westminster Software, Westware, Whitesmith, John Wiley & Sons, Window · XOR Corporation, XQ Software, Xtrasoft · York Street Software · Zylab

For Registration and Hotel Information call today

Produced by Northeast Expositions, 822 Boylston Street, Chestnut Hill, MA 02167 Telephone: 617-739-2000 or 800-841-7000

CIRCLE D204 ON READER CARD

Editor's Note: Part 1, "The Use of a DH11 Multiplexer with RT-11" appeared in Volume 2, Number 5 (September, 1983)

USING EXTRA MEMORY WITH RT-11 AND TSX

or

How to Turbo-Charge RT-11 (Part 2)

By Arthur Edward Groulx, Logicaid Limited, Nepean, Ontario, Canada

Right away the reader will notice that the title contains the word "extra" instead of "extended." The word "extended" conveys the image of some telescoping physical thing, like Pinnochio's nose, or the jaws of the creature in the movie "The Alien." Perhaps this is why most RT-11 users seem afraid of the extended memory monitor (RT11XM) and regard it as a Pandora's box that is better left unopened. This particular article covers the terminology, how extra memory can be used in RT-11 with utilities and tools, and a new tool for greater use of extra memory with RT-11 or TSX on PDP-11/34s.

TERMINOLOGY

There is a huge volume of jargon related to using extra memory with RT-11 and the PDP-11. However, you can safely carry on a conversation with an RT-11 extended memory guru if you know only three terms: physical, virtual, and mapping.

Every character of memory inside your PDP-11 (or LSI-11) computer has a unique label identifying it exactly. This label starts at 0, and increases by one until you run out of memory. This label is referred to as the "physical address." Knowing the physical address, you could pull your memory boards out of the computer and locate the actual chip holding bits in the specific character of memory. The physical address is a name for a physical thing.

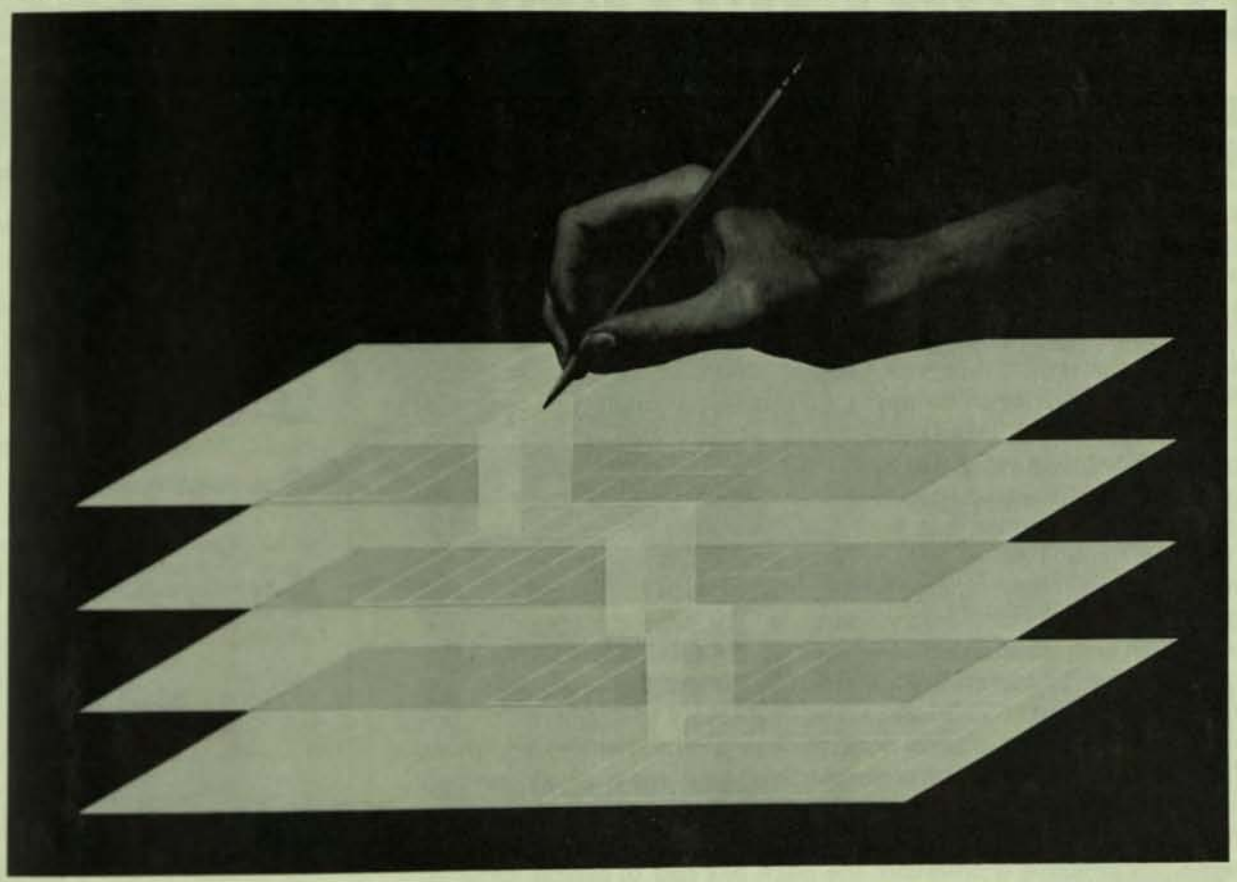
That takes care of physical. Now let's get virtual.

The PDP-11 has what is called a 16-bit architecture. This means when you get down to the bare metal inside the computer, the data values used in the guts of the machine are exactly 16 bits long. In the PDP-11, addresses are treated as any other data values. This implies the labels of the physical things mentioned earlier may be only in the range 0000000000000000 (16 zeroes) to 1111111111111111 (16 ones), in binary, or 0 to 65,535 in decimal. In your MACRO, FORTRAN, DIBOL, DBL, PASCAL, C, or whatever program, at any time, the set of labels that may be used is never larger nor smaller than that set of 65,536 different labels. The 65,536 constraint is referred to as "the 16-bit boundary."

Since the computer can only use 65,536 labels (or addresses), it can remember only 65,536 things; that is, your computer's memory is restricted by that magic number, 65,536. Of course some of this memory is not only remembering information, it's also "remembering" your program. The program, if written correctly, manipulates the remainder of those things. The remainder of those things is usually referred to as data.

Early hardware designers were relatively quick to appreciate the 16-bit boundary, or at least as quick as hardware designers could be. The first attempt at overcoming the constraint was "Hey, let's build a disk — then we can store stuff on the disk and only read in what's needed when we need it." This disk thing was a super concept. We still use

MULTI-DIMENSIONAL MODELING



C-CALC

So Advanced...It's Simple!

DSD Corporation has combined the highest levels of programming technology and human engineering to develop the most powerful...and "simplest" electronic spreadsheet ever.

Easy to Grow Into...

C-CALC meets the needs of both the novice and the experienced user. Menus, informative prompts, and a sophisticated ON-LINE HELP facility allow the user with little or no computer experience to "jump right in". Built-in training procedures facilitate on-going, self-paced learning with tutorial topics ranging from basic to advanced.

And Impossible to Outgrow...

For the experienced user, C-CALC offers a multitude of optional "shortcuts", allowing menus and prompts to be bypassed altogether in many cases. "User-friendly" doesn't have to mean tedious and frustrating to the experienced user. As your applications become increasingly sophisticated, you'll begin to understand why C-CALC has been called the "the most powerful electronic spreadsheet ever"

Features include:

- Menus for ease of use
- Reports that may be edited with wordprocessor or text editor
- Single level English-like commands
- Written in the extremely efficient "C" language (No "C" compiler required on host system)
- Built-in training and sophisticated On-Line HELP
- Accessibility of data on other worksheets
- A wide range of financial, statistical, mathematical, and logical functions
- Data fully transportable between most computers
- Boardroom-quality report generation via convenient print menus
- Dynamic sorting capability
- User-defined procedures for automation of repetitive operations
- Consolidation of like or unlike worksheets
- Easy-to-use data importing/exporting facility
- Multi-level training classes available at our site or yours. Onsite and dial-up demonstrations available

Currently available for:
P/OS (DEC's PRO 350), RSTS, RSX, VMS, VAX, UNIX, and IAS

For more information, please contact:

DIGITEC
DSD
CORPORATION

(206) 821-7507

12620 120th Ave NE, Suite 201,
Kirkland, WA 98033

UNIX is a Reg. TM of Bell Labs
P/OS (DEC's PRO 350), RSTS, RSX, VMS, VAX, and IAS are Reg. TM's of Digital Equip. Corp.

CIRCLE D186 ON READER CARD

we need it." This disk thing was a super concept. We still use them. Some software genius thought of storing programs on disk too and only reading in those programs when needed. This was the invention of the overlay. Every single computer operating system that involves a disk or tape, including RT-11, is really thousands and thousands of overlays. Think about it.

The unfortunate part is that no matter how fast the disk turned on its spindle or how fast the read/write mechanism (heads) glided over the disk surface, the disks were still very slow compared to the speed of the processor itself and the speed with which data could be moved around memory. As far as overlaying went, the central processor spent a good deal of time making Zs waiting for the next overlay to be read in. The entire computer's speed was bounded by the slowest disk being used.

This necessitated the invention of a new technique for speeding up the computer. The new technique would have to be transparent and retrofitable to preexisting software. The intention was to minimize the slow disk accesses.

The eventual solution was to build a "black box" which was placed between the current program and the memory and which intercepted the normal retrieval or storage of the contents of memory addresses. The program presented the black box with an address and said "give me the contents of this address," or presented the box with an address and value and said "store this value at that address." If the black box had 65,536 characters of memory to deal with, the addresses and values would be passed right through to the memory as it was before. There would be a one-to-one correspondence between each address and each memory location.

However, what if the black box had more than 65,536 characters of memory to deal with? In this desirable situation, the black box had to exhibit some intelligence and decide, when presented with an address, exactly which character of memory you were referring to. Remember: the PDP-11 is a 16-bit machine. The address you pass to the black box has exactly 16 bits. But since the black box deals with more than 65,536 characters of memory, it needs a number longer than 16 bits for addressing.

The trick was to make the black box a little computer in its own right. It could take the address you passed to it, perform some arithmetic on it, increase the number of bits in it, and pass the longer address on to the memory to store or retrieve your data. The rules for performing the arithmetic could also actually be programmed into the black box by your own program.

For example, let's say you needed the contents of the address represented by the 16-bit binary number 0 000 010 000 000 (octal 020000 or 8192. decimal). The current state of the black box might convert that address into 0 011 001 010 011 100 101 110 (octal 03123456 or 829230. decimal). By reprogramming the black box, the same address might in the next moment be referring to 0 000 111 111 111 111 111 110 (00777776 octal or 262142. decimal). In these examples, the physical memory addresses used by the black box have 22 bits. The example computer would be said to have "22-bit addressing" even though still only 16 bits can be used by your program. The point is that the same address in your program can at two separate instances refer to two very different places on the physical memory board. To prevent confusion between the address on the physical memory board (see PHYSICAL above) and the address used in your program, the address used in your program was called VIRTUAL. On PDP-11 systems, the black box is called the memory management unit or MMU (it manages the memory), and the arithmetic operations taking place inside the black box are called MAPPING. If the black box is turned on or being used, the system is said to be MAPPED.

A COMPUTER DISTRIBUTOR THAT'S ELIMINATED THE BUGS.

We're First Computer.™ And we've taken the problems out of buying mini/micro computer systems, peripherals and components. Because we not only offer you the finest in computer equipment. Like DEC, Control Data, Emulex, Cipher Data Products and Florida Computer Products. We also offer you the finest in service and customer support.

WE'VE ELIMINATED THE SLOW DELIVERY BUG.

Often you have to wait months to get the computer products you need. But we warehouse a huge inventory of high quality, factory-fresh products. So when you need computer equipment fast, you just have to call us. Or TWX us your order. And we'll pick,

test, pack and ship it within three days. Or the very same day for a small expediting fee. And we can ship complete computer systems within thirty days.

WE'VE ELIMINATED THE NO ANSWER BUG.

When you need information about a product, you want an answer. And you want it fast. But many distributors can't give you technical information. At First Computer, we're experts on the products we sell. And we can tell you exactly how they function. We not only test the individual products. We configure and test your complete system as well. So you can be sure everything you buy from us operates properly. And we continuously examine the viability of new computer products in relation to your needs. So we can help you select products that best meet your application requirements.

WE'VE ELIMINATED THE HIGH-PRICE BUG.

Usually there's only one way to get a good price on computer products. Place a large order. But not when you buy from First Computer. Because we pass along our substantial savings to you. Our prices are some of the most competitive around. And our technical expertise helps you avoid costly errors. First Computer is in business to increase your efficiency.

Reduce your inventory. Give you peace of mind. And, best of all, save you money.

Call First Computer today. And shake the bugs out of your system.

We're humans that function with computer-like efficiency.

First computer corporation

Get the bugs out. Call First now toll-free at 1-800-292-9000 (in Illinois call 1-312-920-1050).

Headquarters: 645 Blackhawk Drive • Westmont, Illinois 60559 • TWX Number 910-651-1916

CIRCLE D210 ON READER CARD

FROM THE RT-11 EDITOR

The programming of the black box is usually done only by systems programmers, and of course the programming must be done with care, but it does not require any more care than other programming. It simply requires the constant remembrance that when the black box is on, every address, including the address of the next instruction in your program, is converted or transformed by the black box. This gets most tricky when in the process of changing the black box, you accidentally change the physical location of the next instruction in your program, and instead of selecting the next physical location in your program (containing the next instruction), you caused the black box to select some other unrelated memory location. Your program will crash, and you will undoubtedly wind up hating the black box, the computer, and go and buy a 32-bit VAX. The process of accidentally killing your program by misprogramming the black box is referred to as "mapping yourself away," as in "I mapped myself away last night." With RT-11 you can change the contents of the black box by using the RT-11 extended memory programmed requests. In the process of doing so, you can also experience the delights of mapping yourself away.

Besides the mapping which transforms a virtual address into a physical address, there is, on some types of PDP-11s, another type of mapping called "UNIBUS mapping." If you care to look at the control register specifications for DEC designed disk and tape peripherals, you will eventually find in some register two bits referred to as "extended memory" or "memory extension." Also, a 16-bit address where the data transfer is to start is placed in some other register. During the data transfer involving the disk, these memory extension bits are appended to the high order part of the 16-bit address placed in the other register, to create an 18-bit physical address of where the data transfer is to physically occur. This is fine if the memory you have is an 18-bit memory. There are problems, however, if the memory you have, like our examples above, uses 22-bits. Apparently you would not be able to transfer information beyond the 18-bit boundary.

UNIBUS mapping is simply the provision of another black box which translates 18-bit addresses in the disk's control registers to 22-bit physical addresses while the transfer is taking place. By programming the UNIBUS mapper, transfers occur in the desired part of memory beyond the 18-bit boundary. UNIBUS mapping does not (yet) occur within RT-11.

At this point you can move on to the next section if you wish, but I thought I would take a few moments to explain two other terms related to extra memory and RT-11: KERNEL and USER.

In the context of extra memory inside PDP-11s, the kernel is not someone who cooks chicken, and the user is not the ignorant twit who uncovers all of your programming errors. In reality, the memory management unit can be thought of as having two black boxes inside. Each

operates independently of the other. When one black box is turned on, you are said to be running in "kernel mode." When the other black box is turned on, you are said to be running in "user mode." Only one of the black boxes can be turned on at any specific moment. Each black box can have its own separate program for changing the virtual addresses into physical addresses. The selection of the current mode is made by specifying some special bits in the Processor Status Word or PS. The neat part is that once you are in user mode, you are highly restricted on how you get back to kernel mode (the hardware won't let you) and, for example, if you try to halt the processor from your program in user mode, you will get back into kernel mode, but only in the way specified the last time you were in kernel mode. Without getting more complicated, the point is that kernel mode is a considerably more privileged mode than user mode.

One final subtle problem: what if you want to look at the memory accessed in user mode while in kernel mode or vice versa? The hardware provides four instructions, MFPD, MFPI, MTPD and MTPI for this very purpose. These instructions move data to or from the "previous" mode's addressing space. The mode of the previous addressing space is also determined by a few bits in the PS. If you examine the program RMON.MAC in the RT-11 distribution kit, you will find a few occurrences of these instructions. If you need to know how to use them, consult the PDP-11 Architecture Handbook, order code EB-23657-18.

RT-11 AND EXTRA MEMORY

The reason for using extra memory is to speed up the computer. One might naturally ask why it is necessary to do this with a single user operating system like RT-11. There are a number of answers.

First, RT-11 isn't always as fast as we'd like it to be. Watch an RT-11 user use RT-11 some time. That user spends a good deal of time waiting for his keyboard monitor commands to be executed. If that wait time were decreased, the RT-11 user would be more productive. Next, some applications are so large that overlaying the programs also decreases the computer's productivity. Some data gathering applications cannot wait for the overlaying. These applications can only be speeded up using the extra memory. Finally, some applications require huge arrays of randomly accessed data which must be continuously memory resident.

RT-11 provides four fundamental modes of accessing the extra memory: virtual jobs, virtual overlays, extended memory programmed requests, and a memory disk called VM.

A. Virtual Jobs

The term "virtual job" describes a special type of RT-11 job which is started and run in a special way. Virtual jobs

. . . continued on page 100

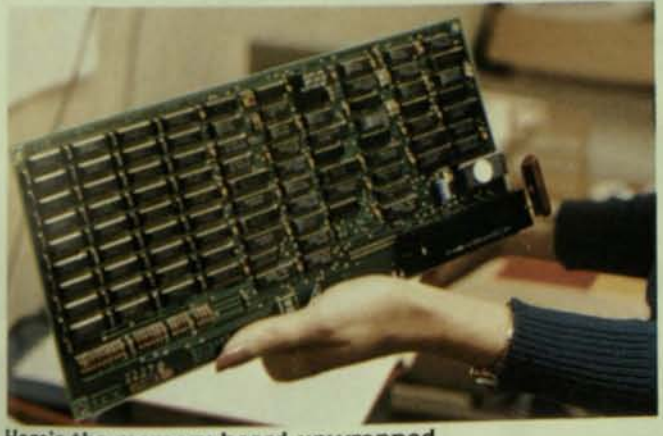
INSTALLING THE 1.7 HARDWARE UPGRADE ON A PRO/350

The components; a memory board and FPP chip.

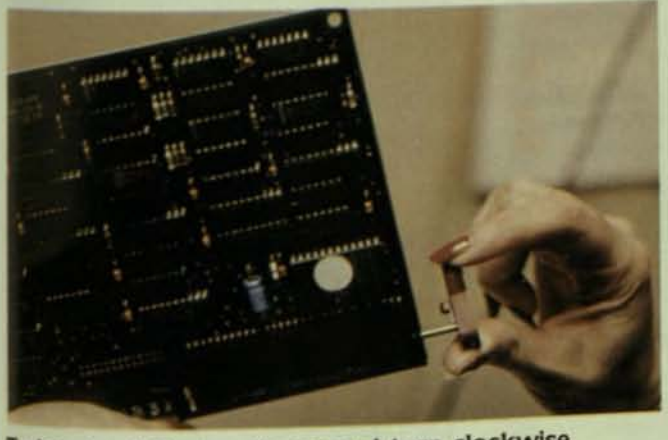
First, remove the cables.

Then, remove the lid.

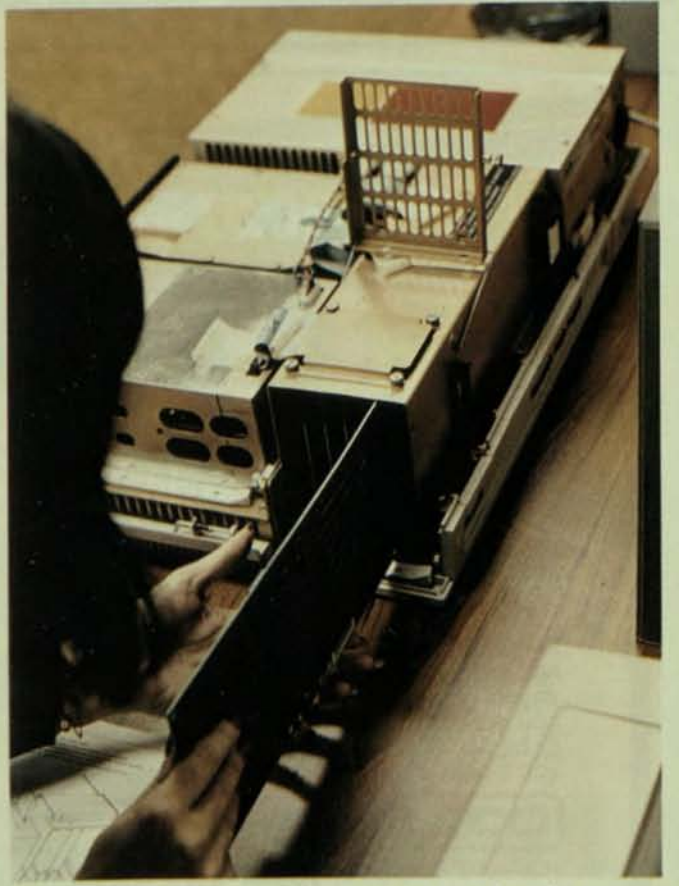
The card cage has a thumb screw on the right — loosen and lift.



Here's the memory board unwrapped.



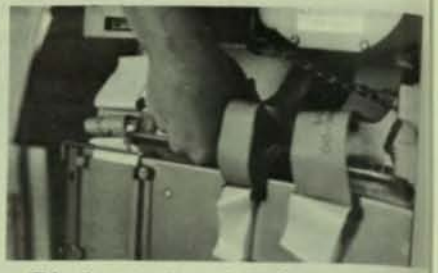
To insert pull the handle out and turn clockwise. This spreads the contacts.



Slide it into any open slot, rotate the handle counter clockwise to engage contacts, and push handle in.

Now for the Floating Point Adaptor.

and the floppy cable (not necessary to remove 3 conductor power lines on left).



This allows you to reach the 3 screws holding the card cage to the main chassis.

Push down on catch with screwdriver (or ball point for purists!).

Disconnect the power connector.

Remove the winchester cables (2)...

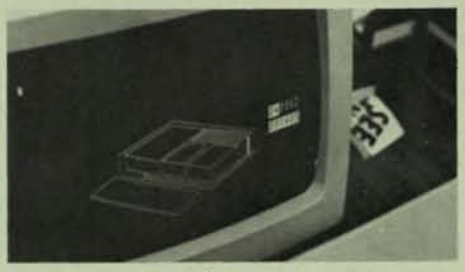
Likewise, disengage floppy catch. Pull both drives out about 5".

Slide the CPU and card cage out from the rear.

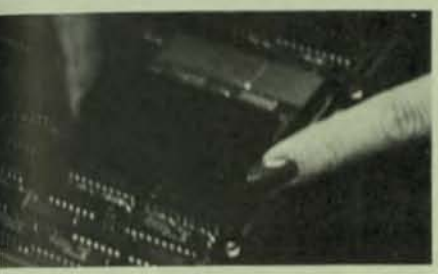
This allows access to the socket for the FPP

... which is next to the CPU chip.

. . . disc cables.



. . . and more errors

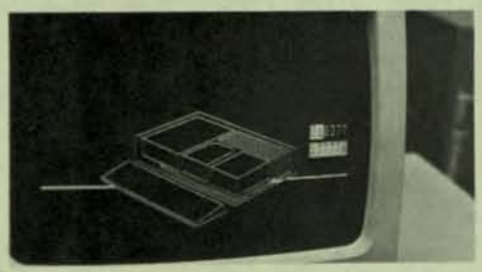


Line up the chip (covered by a black static protector) and push down on the two end tabs.

Close up.

. . . and still more errors.

The cover pops off and the chip is seated.



And get errors ...

Go back and re-seat all cards and connectors and get a LOGO for your trouble.

Save the cover.

Now, re-trace your steps.

Re-connect power . . .

AT LAST, A HIGH PERFORMANC RELATIONAL DATA BASE SYST FOR YOUR PRODUCTION- ORIENTED VAX ENVIRONMENT

You've come to expect high levels of performance from your DEC™ VAX™ hardware. And you should expect no less from your data base software.

That's why Cincom introduced the ULTRA INTERACTIVE DATA BASE SYSTEM™—a next generation relational technology that combines high performance with practical, ease of use implementation.

New Generation Architecture

The secret behind ULTRA's interactive high performance is its unique architecture. An architecture built on a solid foundation of the very latest in relational subject data base technology. In-line control and security are achieved through the ULTRA In-Line Directory while relational implementation is achieved through the unique ULTRA Logical User View and the new Data Base Management facility. Combined, these components provide interactive synergism between users, programmers and the data they wish to access.

For multiple-machine applications that can range from scientific to high volume transaction oriented business systems, ULTRA provides VAX users the control and consistency needed to build high performance, low overhead, high quality systems that can be readily used by both programmers and by the burgeoning number of end-users. ULTRA also facilitates the latest in subject data base strategies for optimum data storage, retrieval and data manipulation.

Integrated Components

The integrated components which provide this unparalleled power include:

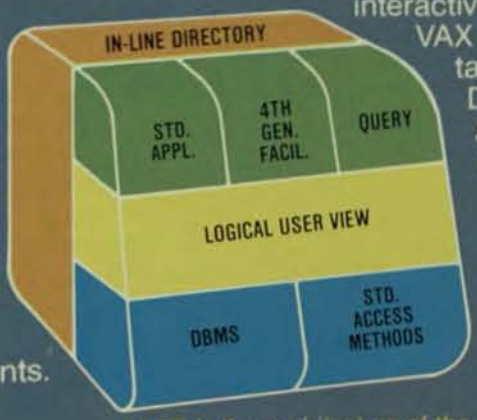
- **The ULTRA Logical User View.** Modern, relational technology to insulate users and application programs from data structures and navigation requirements.
- **The ULTRA In-Line Directory.** Centralized control during program execution. Integration. Security.
- **The ULTRA DBMS.** High performance. Powerful. Faultless non-stop processing. Supports normalized data structures.
- **The ULTRA Intelligent Query and Reporter.** Direct end-user access. Flexible and easy to use.
- **The ULTRA Data Manipulation Language.** So powerful even the most complex processing can be handled using four commands: GET, INSERT, UPDATE, DELETE.

The Ultimate In Interactive Data Base Software For The VAX Environment

Like no other data base system available today ULTRA enables you to achieve the full potential of your VAX system. As your company evolves, ULTRA keeps right in step. And ULTRA is completely compatible with TIS™, our next generation data base system for the IBM mainframe.

So now you can implement Cincom's "IBM-DEC Connection" to link your strategic and tactical applications. In fact, the compatibility is so close, applications built on your IBM can be modified to run on VAX, or vice-versa. An ULTRA/TIS combination is the perfect IBM-DEC Connection.

For an in-depth brochure on the ultimate in interactive data base software for the VAX environment, simply contact our Marketing Services Department, 2300 Montana Ave., Cincinnati, OH 45211, or call us direct.



ULTRA, the architecture of the next generation of interactive data base technology for the VAX environment.

800-543-3010

(In Ohio: 513-661-6000)
(In Canada: 416-279-4220)

Cincom Systems

FROM THE RT-11 EDITOR

. . . continued from page 93

run with the virtual attribute only under an RT-11 extended memory monitor. To mark a job as virtual, location 44 in the job's .SAV or .REL file must have its bit 10 set to one before the job is run. The easiest way to do this is by running the SIPP or PAT utilities as follows:

```
.R PATCH

FILE NAME —
*PROGRM.SAV/A
44/  0  2000
*E
```

Under RT-11 version 4, any save modules so patched will run only under an extended memory monitor. Under RT-11 version 5, patched save modules will run under any monitor.

Making a job virtual does not make it run faster. There is only one major advantage to using virtual jobs, and that is that protection is provided which prevents the virtual job from accessing (and possibly corrupting) the operating system or other programs. A comparison between virtual and non-virtual jobs is provided in table 4-3 of the current RT-11 Software Support manual. The utility KEX.SAV is the only virtual job provided with distributed RT-11. KEX is a virtual job version of KED and runs faster in an RT11XM environment where there are large foreground or system jobs running concurrently.

B. Virtual Overlays

Before the availability of extra memory, RT-11 overlaying occurred by reading sections of the running program into an area of shared memory from disk when the program section was needed. If a main program frequently used two subroutines which overlaid one another, the number of disk accesses would make the program run very slowly.

When virtual overlays are used, all subroutines are already in memory and do not need to be constantly reread from the disk file containing the

original program. The scenario develops as follows. When the program is loaded, the main program is placed in a specific area of physical memory. Subroutines 1 and 2 are placed in their own specific areas of memory as well. Just before subroutine 1 is called, the contents of the memory management unit is changed so that the virtual address used in the CALL instruction is translated correctly to subroutine 1's physical location in memory. Subroutine 1 executes and returns to the main program. Just before subroutine 2 is called, the contents of the memory management unit is again changed so that the virtual address used in the CALL instruction to subroutine 2 now translates to subroutine 2's location in physical memory. Subroutine 2 executes and returns to the main program, and so on. The time required to modify the contents of the memory management unit is extremely small compared to that of reading in the new overlay from disk.

Consequently a virtually overlaid program is extremely fast compared to a traditionally overlaid version.

In fact, in a benchmark we ran, a program was fifty (that's right, 50) times faster when using virtual overlays as opposed to using the traditional disk overlays. Therefore, if you have extra memory available, you should ALWAYS use virtual overlays when overlaying. Use the /V switch instead of the /O switch in your link specifications.

C. Extended Memory Programmed Requests

The extended memory programmed requests are described in section 4.6 of the RT-11 software support manual. These programmed requests are used when virtual overlays are not suitable.

There are two main concepts to understand when using RT-11 extended memory programmed requests: REGION and WINDOW. The term "region" refers to any contiguous piece of physical memory. It is always memory beyond the 16-bit boundary. The term "window" refers to the contents of the memory management unit at any one moment. By moving or "remapping" a window, the physical address corresponding to a program's virtual address is changed. RT-11 provides programmed requests to create and delete regions and windows, and to move or remap windows. A description of the use of these programmed requests is really beyond the scope of this article, as they are normally used in only the most complex systems.

D. The Memory Disk — VM

VM is a handler distributed with RT-11 version 5 which allows you to treat any extra memory as another RT-11 directory-structured device. The VM device is the simplest and easiest way to use extra memory with RT-11. You can create and delete files on the VM device and even use it as the system device with the single job or foreground/background monitor. Unfortunately, you cannot use the VM device as the system device with the RT-11 extended memory monitor.

The VM handler has some nice features. The first feature is that when the handler is installed, it automatically determines the amount of extra memory available and translates that to the number of blocks in the VM device. Since installation of RT-11 handlers is automatic when the system is booted, if you install more extra memory in your system, the VM handler will automatically use it the next time the system is booted.

You can automatically set the base address used by the VM device. This is done by using the

.SET VM BASE = nnnn

command. The base address is a way of specifying the bottom boundary of the physical memory to be used by VM. For the single job monitor, the value of nnnn is normally 1600. For the XM monitor, the base setting varies, depending on whether any of your programs are virtual jobs, or use virtual overlays or extended memory programmed requests. Section 10.12 of the RT-11 System Support manual contains a table of suggested values in those cases. If the base for VM is set too low, then running a job which accesses extended memory results in the error message

?KMON-U-Insufficient memory for region

being displayed when the program is run.

If you have 22-bit addressing (a PDP-11/23+ or PDP-11/70, for example), the VM handler will automatically use the extra memory. If you use the RT-11 extended memory monitor and virtual jobs or overlays or extended memory programmed requests, the value of nnnn in the SET VM BASE command should be 10000 in 22-bit addressing systems.

Applications using a VM type of device will be described briefly in the next section of this article.

As a piece of programming, the VM handler is practically a work of art. (Those readers not interested in RT-11 device handlers in general may go on to the next section without losing continuity.) The VM.MAC file is contained in the RT-11 distribution. It is undocumented, as are all of the other distributed source programs. In examining it, a couple of neat tricks involving RT-11 handlers in general can be seen.

First, during the execution of the RT-11 keyboard monitor INSTALL command, the size of the device being installed may be returned in location 54. In the VM installation code, successively higher addresses are accessed until a non-existent memory trap occurs. The number of blocks in VM are returned in location 54. Later, when VM is loaded and used, the VM handler goes through the $PNAME table in the monitor and searches for the radix-50 equivalent of 'VM' and finds the current size of the device. This technique works for both versions 4 and 5 of RT-11.

Second, the SET VM BASE command actually modifies the disk image of the VM handler. During the execution of the RT-11 keyboard monitor INSTALL command, channel 17 (octal) is open to the device handler's disk image. The SET VM BASE command changes the disk image. The handler

must be reinstalled so the new base can be used in the installation automatic sizing code.

VN: ANOTHER MEMORY DISK

If you have a computer that supports 22-bit addressing, you can really load up your system with memory and have a very large and fast system device. Eligible computers are the PDP-11/23 PLUS, the PDP-11/44, the PDP-11/70, and the new processors containing the J-11 chip, the PDP-11/75, the PDP-11/83 and the PDP-11/84. You can use the VM handler described above.

If, like tens of thousands of RT-11 users, your computer is a PDP-11/34, you don't have to be excluded from the 22-bit addressing world. There is an extra hardware device you can purchase which permits 22-bit addressing on that formerly 18-bit addressing computer. This device is called the ENABLE/34 and it is manufactured and marketed by ABLE Computer of Irvine, California. The PDP-11/34 was the popular workhorse of the PDP-11 series. Used 34s can now be obtained for almost the price of shipping. Adding the ENABLE to the 34 changes it into a very fast development system using the more modern 22-bit technology.

In order to use the ENABLE/34 on a PDP-11/34 with RT-11, you need three things: the ENABLE/34 board and enough backplane on which to install it, a memory board containing 22-bit memory, and a piece of software called the VN handler. The VN handler is developed, marketed, and supported by Logicaid Limited of Nepean, Ontario, Canada, the company for which I coincidentally work. The VN handler is fully TSX compatible and can be used to speed up TSX on the PDP-11/34 as well. The VN handler works exactly like the VM handler, except it is also bootable with an RT11XM monitor.

For the purposes of developing the VN handler and testing to see whether it was viable with RT-11 and TSX, ABLE computer provided us with an ENABLE/34 and an associated device called an ENABLE/CACHE. The cache memory is another way of turbocharging the 11/34. A cache memory is a very fast piece of memory which contains the contents of the last memory location accessed, and the 8,191 characters following that memory location. When the processor requests the contents of a memory location, the cache intercepts the request and first checks to see if that location is in the cache. If it is, it very quickly returns the contents to the processor. If it isn't, the request is transparently passed along to the main memory and the cache starts loading itself from the requested address. The theory is that most programs are used in a very local manner. Instructions and data are grouped together, and so tend to reside in the cache together. This theory is confirmed by the fact that when RT-11 is running by itself, the main memory is never accessed. Only the cache is used. This is because the RT-11 monitor easily fits completely into the 8,192 character cache.

When the ENABLE/34 is installed, the main constraint is that all 18-bit controllers must reside in front of it on the bus. Without boring you with the details, this usually means that boards must be shuffled. Both the ENABLE/34 manuals and ABLE personnel are extremely helpful in this regard.

After the boards were installed, the VN handler was tested. Like the VM handler, the VN handler will automatically determine how much 22-bit memory is available. On the test 11/34, we used a MOSTEK 1-megabyte memory card. Unlike the VM handler, the VN handler does not use the PDP-11 memory management unit. The advantage of this is that programs which run under the RT-11 extended memory monitor are not adversely affected, and the VM handler can be used exactly as before.

When the VN device was initialized, it correctly showed that it had 1,522. blocks of "disk" space available. This is over 1.5 times the size of a dual density floppy, and of course incomparably faster. If we had a 4-megabyte memory board, the VN handler would give us a memory disk with 7,666. blocks of memory disk space. This is large enough to run most applications.

Attn: DEC OEMs, ISOs and End Users

SOFTWARE YOU CAN COUNT ON

From Theta Business Systems

Six Integrated Financial Packages

Over the past 4 years, Theta Business Systems has written and supported the most reliable, user-friendly software in the DEC marketplace. Established in 1979, the founding team at TBS was a group of consultants each with years of experience, working independently with small and medium sized businesses. Their combined familiarity has guaranteed an informed attitude toward the end user. Whatever you need, whether it's RT-11, RSX-11M, RSTS/E, VMS or CP/M 86, we have the accounting software, support and prices you've been looking for.

In a recent independent survey (unsolicited by TBS), 90% of our users were very happy with the overall performance of our business software and support. This exceptional user satisfaction says that you can have confidence in your installations.
David Karas, OEM

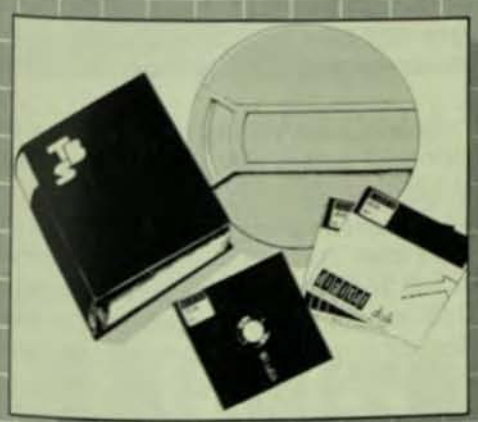
"I especially like the parameter files for screen and file definitions and reports. I brought up my customer without ever recompiling the programs. I was really impressed."
Armadale Co., Ltd.

"Armadale has found the TBS packages to be very well developed and user oriented. Since the sources are supplied, the individual packages can be modified to accomodate the users' needs. Overall, we are pleasantly surprised with the versatility of the systems."
Jack Mulchinock, Systems Mgr.
Armadale, Co., Ltd.

"TBS is responsible. John Cumming (Director of Field Support) is an excellent person to have as a resource. The programs are well documented and extremely complete. TBS is a responsive support organization. I'm very happy with things."
Daniel Fishman, VP/Gen. Mgr.
NDC Systems

"The staff that are using it find the interactive nature of the systems quite easy to work with — easy to train people to use it. I'm quite pleased with support from the standpoint of having easy access to the technical support people. One of the strong things is that source is supplied; that allows the user to control and deal with their own system. The system is working quite well for us."
Evan Friedman, President
California Data Corp.

- **Payroll**
- **Accounts Receiveable**
- **Accounts Payable**
- **General Ledger**
- **Order Entry with Inventory Control, Billing and Sales Analysis**
- **Time Accounting for Professionals**



Documentation manuals may be purchased for $35 per system. Details of each module include user instructions, data entry summary, screen ad report layouts, file definitions, and I/O diagrams. A full feature demo is available for $60 per system (e.g. G/L) plus media cost. (Media: magtape, floppies, hard disk) Documentation manual is included. Cost of manual(s) or system demo(s) will be applied to system purchase.

CONTACT US TODAY

For further information, color brochure, product descriptions, full system demos, or manuals — call Chuck Tebbetts, Director of Marketing 213/245-0917 or 213/242-7981.

THETA BUSINESS SYSTEMS

1110 Sonora Avenue, Suite 106
Glendale, California 91201

CIRCLE D145 ON READER CARD

FROM THE RT-11 EDITOR

After insuring the VN handler worked perfectly, we ran a set of sixteen benchmarks to compare the speed of a VN-based system with an RL01-based system. There were four types of benchmarks: Sorting, Searching, FORTRAN compiles, and MACRO assemblies. These are fairly common tasks performed in the computing environment. Each benchmark type was run on the VN memory disk with cache, on the VN memory disk without cache, on the RL01 hard disk with the cache, and on the RL01 hard disk without the cache. The following table indicates the results. The numeric table entries have been normalized so that the slowest benchmark of each type corresponds to a value of 100. A value of 76, for example, means that that benchmark took only 76 per cent of the time of the slowest task. The smaller the number, the faster the task.

	Memory Disk with Cache	Memory Disk without Cache	Hard Disk with Cache	Hard Disk without Cache
Searching	11	14	100	100
FORTRAN Compiles	57	76	90	100
MACRO Assemblies	67	89	77	100
Sorting	75	100	81	85

In most cases the memory disk with cache is significantly faster than the hard disk. The turbocharging effect of the cache is obvious in all cases.

For searching applications, the memory disk is considerably faster. A typical application would store any files which frequently need to be searched on the memory disk. A significant improvement in system performance would be obtained. DIBOL or DBL ISAM key files fall into this category.

The only thing about the results that puzzled us for a while was the fact that sorting was slower on the memory disk without the cache. For sorting we used the ZORT program product. The slower memory disk figures are a consequence of the fact that ZORT is so efficient in terms of overlapping disk I/O and processor usage. When using the memory disk, I/O cannot be overlapped with processor usage as the processor is involved in doing the memory disk I/O. When a program runs slower with VN than it does otherwise, it is because there is a high level of overlapping of disk accesses with processor usage.

Our final conclusion is that the use of 22-bit memory on the PDP-11/34 with RT-11 or TSX is viable and can dramatically increase system performance. It should be considered as another turbocharger for your RT-11 based PDP-11/34.

UNITRONIX represents DIGITAL

PDP 11/23 SYSTEMS with 256kB* memory, 160 Mbyte winchester, std. 45 ips 9-track tape backup, VT101 CRT, LA120 printer. Runs RSX11, RSTS, TSX OR UNIX. $32,000 *up to 4MB avail.

TERMINALS
VT100 LA12
VT101 LA34
VT102 LA50
VT125 LA100
VT131 LA120

CPUs
PDP 11/23, 11/23+, 11/24, 11/44 & VAX

PCs

INTERFACES
MODEMS

COMPLETE DATA SYSTEMS
with Unitronix-developed standard or customized applications software.

VT, LA, LSI, PDP, VAX are trademarks of Digital Equipment Corporation

DEALERS WANTED . . .
WITH COMPUTER SERVICE CAPABILITIES

WHY YOU SHOULD CALL UNITRONIX FIRST

- **Immediate Delivery from $6 mm Warehouse Inventory**
- **Complete Equipment Service Capability**
- **Excellent Pricing/Discount Structures**

In just over 6 years, Unitronix Corporation has emerged as a leading supplier of Digital Equipment Corporation (DEC) microcomputers, terminal products, printers, accessories and supplies on both a domestic and international level. Our new, ultra-modern 20,000 sq. ft. facility in Somerville, New Jersey, houses executive offices, in-house programming, customer service, sales and marketing, component testing and complete warehouse facilities.

UNITRONIX CORPORATION
197 Meister Ave., Somerville, NJ 08876
TELEX: 833184

AUTHORIZED digital® TERMINALS DISTRIBUTOR

CALL FOR LATEST PRICES

(201) 231-9400

CIRCLE D154 ON READER CARD

Editor's Note: John Gram in his continuing series of articles, "The Elements of Software Support," has focused on "Understanding the Elements of Support"(Vol. 2, No. 2); "The Softness of Support"(Vol. 2 No. 3); "Planning and Preparing For It"(Vol. 2 No. 4); and "Managing the Support Effort"(Vol. 2 No. 5). The fifth in this series addresses the controversial area of customer support.

Has the technology that gave us availability authored a new code of business ethics? Does the customer compromise his integrity in the marketplace if he fails to police industry-wide software distribution?

Mr. Gram's view is his own and does not in any way reflect that of THE DEC* PROFESSIONAL.

AN OPINION:
ELEMENTS OF SOFTWARE SUPPORT: THE CUSTOMER'S RESPONSIBILITY

By John Gram, Placentia, CA

In this series on software support we have discussed the responsibilities of those creating and of those distributing software products. This article addresses their roles and responsibilities. What responsibilities must the customer assume? First, customers must realize that some very real costs are associated with providing product support. Next, customers should know that they must bear the cost of this support. Mental assent must result in a willingness to purchase these services.

But let's back up a few steps. By the time most customers begin to think about support they have already neglected what should have been their first responsibility: careful research of available products, feasibility studies and eventual purchase recommendation. The final recommendation must then be evaluated for accuracy, objectivity, thoroughness, and expansion for future plans. This process, slow though it may be, usually results in the purchase of the best product.

Many problems are eliminated when a customer follows through on this responsibility. Responsible evaluation can prevent a software **bargain** from turning into a software **disaster**. For instance, some companies may offer software at lower prices, but some of these same companies do not offer support services of any kind. The customer has saved no money if the product **or** services are inferior. Sev-

eral considerations must be taken into account: price; functionality; reliability; performance; adequate documentation, training and support services; stability and viability of the company; reputation of the company; operator satisfaction; and satisfaction of existing customers. Feasibility studies must determine the answer to these questions.

After choosing a product, the customer must assume the responsibility of integrity. As computing systems proliferated in the marketplace, their prices have decreased. As payroll costs have grown, the price of developing (and supporting) good software has increased. Computer companies used to give their software away to encourage the purchase of equipment. This trend has disappeared. In fact, with the price of developing software, it seems as though software costs are beginning to exceed equipment prices. Unfortunately, software is fairly intangible to many — especially to those unfamiliar with computing and the labor intensive job of creating good software. The temptation to share software within the same company or even to extend it to another is tempting. Some are even ignorant to the fact that this sharing is outright theft. Very often those involved intend to arrange additional licensing. After mentally committing to purchase the licensing, they install the software. When the crisis that demanded the software's presence disappears, it becomes easy to overlook the technicalities of licensing. In fact, it becomes easy to assume that another individual in the organization has handled it. While this oversight is understandable and rational, integrity dictates that closer attention be paid to these issues.

Customers reasonably expect to have software products supported; and vendors reasonably expect to be paid for their software development and support. Customers must assume responsibilities to pay for the software and support they consume. Even the telephone utilities are beginning to base charges on usage. Should a software vendor go out of business, the value of the product to its user becomes questionable. In a sense, the penalty for unauthorized distribution may be paid by those with rights to the software, while those outside of their legal rights are contributing to the downfall of the software provider.

High speed, high quality copiers have turned virtually every corporation into a potential publisher. Similarly, the affordable computer enables every company possessing a computer to become a distributor of software. The illegal distribution of software usually occurs without the knowledge of principals within these corporations. The call to integrity must include general communication of corporate policy on such illegal distribution to all employees. Many licensing agreements include a statement that the licensee will do everything within his power to restrict illegal distribution. This includes communicating the license terms to those who have access to the software.

Customers must recognize that they are not purchasing a software product. They only buy the right to use it. The product they are using is too expensive to throw away for a new one when it stops meeting their needs. Purchasers of computing systems must begin to recognize that they must pay reasonable maintenance fees to keep their machinery running smoothly. Corporations are entirely free to avoid maintenance fees. However, these same corporations must

realize that the software vendor cannot provide service when payment has not been made. Should the customer want to reinstate service, higher fees are not unreasonable. Some have argued that charging a higher fee to regain supported status is a form of blackmail to assure constant support. While this scheme could certainly be used for blackmail, it is also a reasonable and fair method of amortizing development costs. For the support process to work best, communication is the key. Customers must first communicate clearly to their potential vendors that they **expect** maintenance. They must clearly communicate that they recognize the cost and are willing to pay for the support service. After these initial requirements have been met, the actual work of support begins.

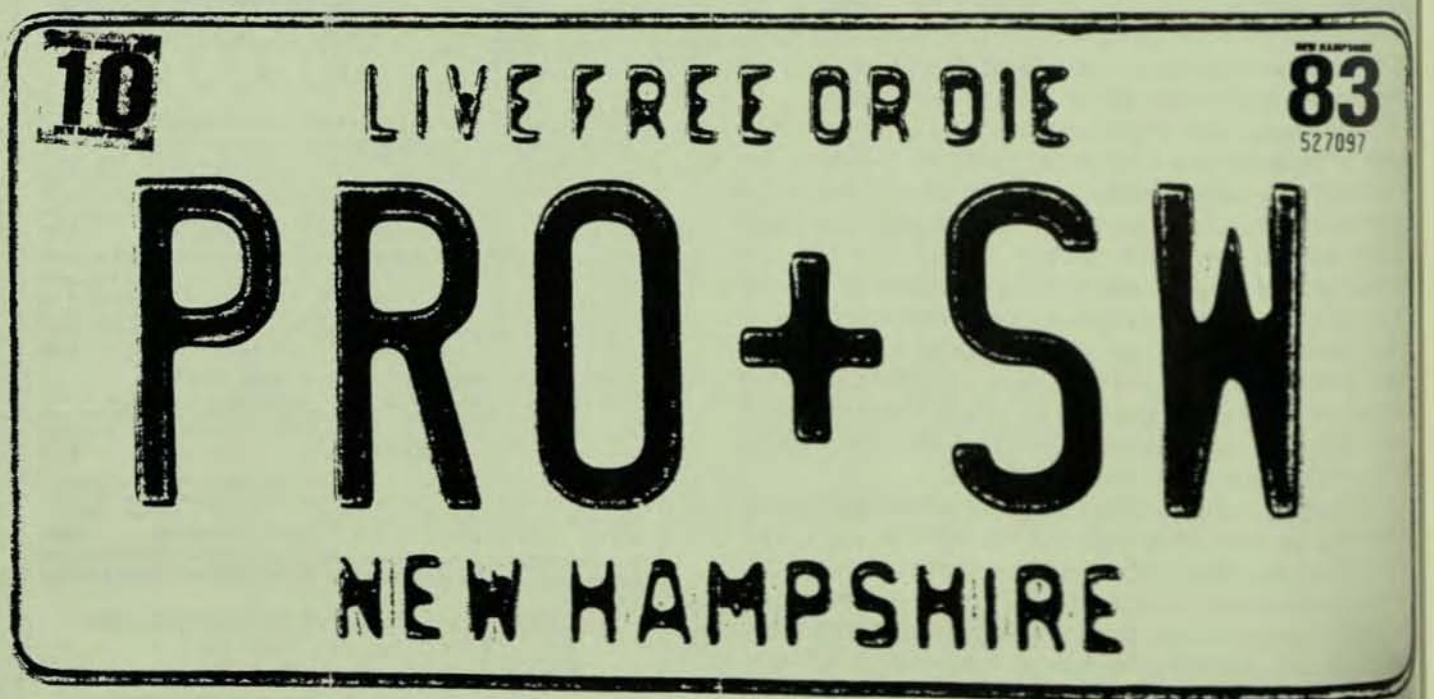
Clearly, **communication** requires both a clear **presentation** and a clear **reception** of facts and emotions. While the support effort requires much more than communication, it is usually communication that determines its apparent success or failure.

Customers must understand that software can only be affordably maintained when they are willing to wait for enhancements. Unfortunately, the age of computers has given people the excuse to be demanding, unforgiving, and impatient. Computers perform at tremendous speeds. Production requires these computers to be working all of the time. Tremendous pressures are placed on all of us in the computing industry. I wouldn't be surprised if life insurance companies start charging higher premiums for those in the computing field.

We all know there are times when hardware or software repairs are urgent. Consumers: Educate yourselves to recognize the difference between highly desirable and essential changes. Recognize which problems can be worked around and which cannot. Consider the viewpoint of the average customer of the product. If a request ends up outside the scope of common product usage, realize that a longer wait for a solution may be reasonable. Perhaps additional charges to expedite implementation may not be totally unreasonable. In general, the price paid for support directly relates to the depth and timeliness of support received.

I have found that customers are willing to wait when past performance has shown a solution **will** come. When solutions are too long in coming, there is a greater tendency to "cry wolf." Do not be surprised if a response to unreasonable demands does not occur. A very large contributing factor to this lack of patience among computing professionals is the lack of maturity. Because the dramatic increase of computing has occurred in the very recent past, the average age of those involved is often lower than for other fields. Because of the shortage of available professionals, and the difficulty in living without a computer professional for even a few days, lower levels of maturity are sometimes tolerated when they might not otherwise be. The artificial power that computer knowledge and control give the professional can have an inebriating effect. While this is true at any age, youth seems more susceptible.

Purchasers of software must be wise in their shopping. They must be aware of the need for support services and willing to pay for them. And they must be able to fairly evaluate exactly how much support they have purchased. Finally, they must mix maturity, reason, and communication skills with those of the supporting agency to result in a high level, professional service. Only then are high expectations reasonable — and reachable.



Develcon introduces

DEVELSWITCH

An advanced intelligent switching system for people who want it all.

Of course you deserve it all. And to satisfy your desire for performance, we're introducing the Model 9006 Develswitch, a technologically superior intelligent data switching system. Once you learn more about it, you'll find that Develswitch stands out from all others with its unique combination of advantages that reduce costs, provide extraordinarily broad applications flexibility, speed communications, enhance convenience and allow for easy growth. If you really want it all for your datacomm network, now you can have it with Develswitch.

Develcon ELECTRONICS

Develswitch Model 9006 Intelligent Data Switching System — Features/Advantages

The Model 9006 represents the current state-of-the-art in intelligent data switching systems, and the logical evolution of our many years of switching experience. With automatic port contention and selection for thousands of synchronous or asynchronous conversations, the 9006 incorporates sophisticated software enhancements which provide a high level of system customization features never before available, along with many new features that offer unprecedented performance advantages. With Develswitch, now you can have it all. Ask us today for technical literature or applications assistance.

Consider these Develswitch features:

- CONTENTION AND SELECTION for asynchronous and synchronous lines to speeds of 19.2 Kbps
- ADVANCED SYSTEM SOFTWARE increases applications flexibility, speeds communications, reduces costs
- NETWORK SUPPORT for up to 2000 lines per switch
- 16 USER-PROGRAMMABLE SYSTEM PROMPTS allow customization to meet specific application requirements
- PRIORITIZED QUEUEING with update and exit options
- FIELD-UPGRADABLE MODULES provide virtually unlimited growth capabilities
- SPEED SELECTABLE FORCE CONNECTIONS
- MULTIPLE PROCESSOR CONTROLLERS to accommodate even the largest networks
- OPTIONAL REDUNDANT POWER SUPPLY
- ENHANCED SYSTEM OPERATION DISPLAYS show complete and instantaneous system status
- AUTOBAUD OR CONFIGURED speed/code/parity connections
- PARITY LINE CONNECTIONS
- MESSAGE FORWARDING provides current connect information
- CONFIGURATION MEMORY backed up by battery
- ACU SUPPORT
- TANDEM LINK HARDWARE/SOFTWARE connects two or more intelligent data switching systems
- 5 Mbps THROUGHPUT
- SYNCHRONOUS SELECTION via X.21 protocol or Keypad Select Unit
- DOUBLE SAMPLING via operator command
- TRUE AUTOMATIC SWITCHOVER to redundant common logic
- SPECIAL SOFTWARE ENGINEERING SUPPORT through our customized software services to meet your precise application
- 8 CHARACTER PROGRAMMABLE PASSWORDS AND DIRECTORIES
- SELECTABLE TIME-OUT INTERVALS
- SINGLE OR TRIPLE BREAK DISCONNECT
- 43401 OR HIGH LEVEL Limited Distance Data Set line interface cards
- CHOICE OF RS-232, barrier strip and mass terminated backpanels
- INDIVIDUAL DIRECT LINE-TO-LINE ACCESS
- FULL DIAGNOSTICS including over 70 error messages, over 20 statistics and on-line validation of firmware
- SUPPORTS ALL COMMON CHARACTER SETS including ASCII, EBCDIC, BCDIC
- COMPATIBLE WITH ALL DEVELCON intelligent switching and data transmission products
- ONE-YEAR WARRANTY, NATIONWIDE SERVICE

Develcon ELECTRONICS

DEVELCON ELECTRONICS, INC.
4037 Swamp Road
Doylestown, PA 18901
Telephone: (215) 348-1900
Telex: 84-6389

Develcon data transmission products . . . cream always goes to the top.

CIRCLE 0200 ON READER CARD

Develcon Electronics is a leader in
We are manufacturers of a complete line

FRECEs
A NOTE ON FREE RECORD CHAINS

By Charles Mansfield

Freak out over free record chains? FRECEs are fast and fascinating for filling fixed-length files. The principle can be applied to client files or large, fairly static, transaction files where deleted records must be found quickly and re-used. A free record chain is invaluable in a situation where your files cannot be extended easily.

The idea is to keep track of all the free or empty records in a file so that when a new record is added or filled up a space can be found without using any system degrading file handling technique. Your ISAM indices are used only for finding full records, which is what you pay them for. In fact with a free record chain the next free record is always pointed to by the file control record (record one). So only one read is needed to find an unused record.

FRECEs are so good that if you use them, you'll become an overnight star in the dp department.

OK. Let's start at the beginning with a fresh, empty file. I normally fill empty files with hi-values such as ASCII 127s. Then I know what I've initialized and what's disc garbage. Let record one be a file control record with its first field holding the pointer to the next free record. In our fresh file this will be record number two. The first field in record two must point to the next free record and so holds the number three. Record three's first field points to four and so on.

You'll have to write a little initialize program or chain builder to run through the empty file and set all this up. Write two into record one's first field, three into two's, four into three's and so on.

Now imagine that you have to find a free or empty record in an update or data capture program to put a new record of data in. All your program has to do is read in the first record of the file, look at the first field and that will say where the next free record is.

Simple, but remember to tidy up after yourself. Before you overwrite the whole free record with the user's data you must copy the number of the next free record back to the file control record. You must not break the chain, as it were, just reforge the links around the used records.

All well and good. You can imagine the file filling up nicely with the pointer in that first record clicking up one at a time. Then someone wants to delete a record, somewhere in the middle of the nearly full file. Record 30, say, is to be deleted but the pointer in record one says the next free record is 100. What do you do?

When you have deleted the record in your program you have to relink the chain through that newly emptied record. With the newly emptied record in memory at the same time as the first record, just swap (LSET) the numbers held in their first fields. The next addition to the file will then grab that newly emptied record and the previous next free record will be pointed to again.

Hello. Who's still with us? It's all a bit hard to swallow at first reading, but it's worth understanding, so let's try a comparatively wordless explanation. In the list below each digit represents the contents of the first field of each record in a FRECEd out file:

2,3,4,5,6,7,8,9 All freshly initialized.

Add a new record . . .

3,full,4,5,6,7,8,9

Add a new record . . .

4,full,full,5,6,7,8,9

Add a new record . . .

5,full,full,full,6,7,8,9

Delete contents of record three . . .

3,full,5,full,6,7,8,9

Comprendez?

Dear Dr. DEC:

I could use a little help in getting some computers to talk to each other. We have a PDP 11/34 with RSX11-M and several LSI 11/23s. The 11/23s use the MXV11-BF multi-function board populated with an MXV11-B2 boot prom set. This is a humdinger of a combination, touted by DEC to provide bootstrap loading of operating systems and application software from DECnet. Sounds great, since I want to down-line load from the 11/34 to the 11/23s. Now then, how do I make it happen?

The prom user guide says that I can do a DECnet boot from a DLV11. The Tech Bulletin says the serial lines on the MXV11 are software compatible with DLV11s. First Problem: the CSRs aren't where they are supposed to be. In fact, my serial line is at 176500 (nice standard address) but the prom is looking for it at 175610. Well, maybe I can get around that one by booting to a nonstandard CSR.

Now, all I have to do is buy DECnet for the 11/34 and I'm home free, right? Second problem: Nobody seems to know. The DEC experts ask, "What's your operating system on the 11/23?" Now, why would I want to go and screw up a perfectly good 11/23 with an operating system? All I want to do is handle a clock interrupt, read some analog input, and send it back over the DLV11. Why can't I just sit down at the 11/34 and assemble some macro code to work the hardware, link in a little FORTRAN to do some calculations, and downline load the .TSK via DECnet? I'd even be willing to give up the FORTRAN if I had to.

The fact is, most of the things I want to do are already happening on the boot prom. Third problem: who's got the source listing?

We would appreciate any Guru-type help we can get on this one.

Thomas H. Haydon
Section Manager, Electronic Systems
BF Goodrich, Troy, Ohio

* * *

Dear Dr. DEC:

In response to Joseph Heck's letter in the September '83 issue, here is one answer which works. I am writing this letter on an 11/23 with 248Kb of main memory and an RKV11 modified this way. Sure looks like it works! The only warning — the double sided tape must be the thin type or the card thickness will be so great that it will not fit in the backplane without sacrificing the slot immediately in front of the component side.

Keep up the good work. [*See FIGURE A for instructions.*]

David G. LeVine
Nashua, N.H.

* * *

Dear Dr. DEC:

In June '83, I upgraded our system to Version 7.2 of RSTS and Version 2.0 of BASIC PLUS TWO. Soon after the installation I noticed some of our reports had lines that were printing over one another. When I looked in the print file (Sequential Stream) using "Editor" I noticed that there was a CR at the end of the overlapped line, and the line that should have printed under the overlapped line continued on the same line. I took that same program and re-compiled it on another system using 7.2 of RSTS but Version 1.6 of BASIC PLUS TWO and the problem was corrected. I don't think I should have to re-compile every program that uses the "print" statement on another system. The problem is getting worse. I must find a solution to this problem. Can you or any of your readers help me?

Marty Chojnacki
Inland Diesel Inc.
Butler, WI

If that is the only problem you have with V2.0 of BASIC PLUS-2 you are lucky. See if it is repaired in V2.1 which is now available.

* * *

Dear Dr. DEC:

We have a current program that has a large array (10K words) which is

FIGURE A

```
RKV-11D MODIFICATION INSTRUCTIONS FOR 18 BIT ADDRESSING CAPABILITY


1.  BACKPLANE MODIFICATIONS
    1.1 CUT ETCH BETWEEN C1H1 AND GROUND
    1.2 ADD WIRING BETWEEN C1H1 AND C4H1
    1.3 ADD WIRING BETWEEN A1E1 AND C4N2
    1.4 ADD WIRING BETWEEN A1D1 AND C4P2

2.  M7268  MODIFICATIONS
    2.1 COMPONENT SIDE  WIRING
        2.1.1   ADD WIRING BETWEEN E6-4 AND E4-5
        2.1.2   ADD WIRING BETWEEN E5-1 AND E4-4
        2.1.3   ADD WIRING BETWEEN E4-6 AND PIN CH1
    2.2 NON-COMPONENT SIDE  WIRING
        2.2.1   ADD WIRING BETWEEN PIN CN2 AND J1SS
        2.2.2   ADD WIRING BETWEEN PIN CP2 AND J1UU

3.  M7269 MODIFICATIONS
    3.1 ADD A NEW 8881 CALLED E14B
        3.1.1   ADD AN ADDITIONAL DEC 8881 I.C. BY BENDING PINS 1 THROUGH 6
                AND PINS 8 THROUGH 13 INTO A HORIZONTAL PLANE AND MOUNTING
                THIS DEVICE OVER E14 (ALSO A DEC 8881 I.C.) USING DOUBLE SIDED
                TAPE ORIENTING E14B-1 OVER E14-1 AND E14B-7 OVER E14-7 AND
                E14B-14 OVER E14-14 ETC.
        3.1.2   SOLDER E14-7 TO E14B-7
        3.1.3   SOLDER E14-14 TO E14B-14
    3.2 COMPONENT SIDE WIRING
        3.2.1   ADD WIRING BETWEEN R25-3 AND E14B-8 AND E14B-9
        3.2.2   ADD WIRING BETWEEN R25-4 AND E14B-5 AND E14B-6
        3.2.3   ADD WIRING BETWEEN R24-7 AND E14B-2 AND E14B-3 AND E14B-4
        3.2.4   ADD WIRING BETWEEN R24-9 AND E14B-10 AND E14B-11 AND E14B-12
        3.2.5   ADD WIRING BETWEEN E14B-1 AND PIN AD1
        3.2.6   ADD WIRING BETWEEN E14B-13 AND PIN AC1
    3.3 NON-COMPONENT SIDE WIRING
        3.3.1   ADD WIRING BETWEEN J1SS AND R25-4
        3.3.2   ADD WIRING BETWEEN J1UU AND R25-3
```

SPECIAL SUBSCRIPTION OFFER

VAX RSTS PROFESSIONAL

Box 361
Ft. Washington, PA 19034-0361
(215) 542-7008

SPECIAL SUBSCRIPTION OFFER

☐ PAYMENT ENCLOSED for one year's subscription (6 issues).
US: $35. Canada & 1st Class: $50 US. All other countries, air mail: $60, payable in US dollars.

☐ BILL ME for one year's subscription (6 issues). ☐ US ☐ Canada or 1st class ☐ Foreign.

☐ VISA/MasterCard
Account# _____ Exp. Date _____

Signature: _____
(must sign)

Name _____
(please print or type)

Address _____ Suite _____

City _____ State _____ Zip _____

Telephone ( ) _____ Country _____

FREE CLASSIFIED AD WITH SUBSCRIPTION!
Your first 12 words are absolutely **FREE**. Extra words only $1.00 per.

DP31

Not many things can be called "the one and only."

In 1979, we introduced the one and only independent magazine exclusively for RSTS users. Now, we proudly announce the expansion of that internationally-respected journal into the only magazine for both VAX and RSTS users: VAX/RSTS PROFESSIONAL.

That's right. We expanded RSTS PROFESSIONAL into a new, bigger and more complete magazine created exclusively for both VAX and RSTS users.

We did it, not by cutting down on RSTS articles, but by adding articles on VAX to create a bigger and better journal.

So if you're into VAX, then the new VAX/RSTS PROFESSIONAL was created for you. And if you know anything about RSTS PROFESSIONAL, you'll know the high-quality and technical excellence you can expect.

Our articles are by users whose names you'll recognize; users who know DEC hardware and software because they use it each and every day. Tips, insightful product reviews, answers to your questions, expert opinions and more will be inside every issue — all the things that will give VAX and RSTS users that "professional edge."

Subscribe today to the magazine that is indispensable to every savvy VAX and RSTS user.

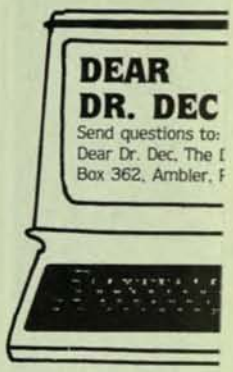
NO OTHER MAGAZINE FOCUSES EXCLUSIVELY ON VAX AND RSTS.

Don't miss even one issue. Fill out the postage-paid subscription card and return it today.
Only $35 for six issues.

VAX RSTS PROFESSIONAL

P.O. Box 361, Fort Washington, PA 19034 ☐ (215) 542-7008

CIRCLE D220 ON READER CARD



Dear Dr. DEC:

I could use a
some computers
We have a PDP
and several LSI 1
the MXV11-BF r
populated with
prom set. This i
combination, t
provide boot
operating syste
software from D
since I want to
the 11/34 to th
how do I make

The prom use
do a DECnet bo
Tech Bulletin says the serial lines on the MXV11 are software compatible with DLV11s. First Problem: the CSRs aren't where they are supposed to be. In fact, my serial line is at 176500 (nice standard address) but the prom is looking for it at 175610. Well, maybe I can get around that one by booting to a nonstandard CSR.

Now, all I have to do is buy DECnet for the 11/34 and I'm home free, right? Second problem: Nobody seems to know. The DEC experts ask, "What's your operating system on the 11/23?" Now, why would I want to go and screw up a perfectly good 11/23 with an operating system? All I want to do is handle a clock interrupt, read some analog input, and send it back over the DLV11. Why can't I just sit down at the 11/34 and assemble some macro code to work the hardware, link in a little FORTRAN to do some calculations, and downline load the .TSK via DECnet? I'd even be willing to give up the FORTRAN if I had to.

The fact is, most of the things I want to do are already happening on the boot prom. Third problem: who's got to Version 7.2 of RSTS and Version 2.0 of BASIC PLUS TWO. Soon after the installation I noticed some of our

FIGURE A

```
RKV-11D MODIFICATION INSTRUCTIONS FOR 18 BIT ADDRESSING CAPABILITY


1.  BACKPLANE MODIFICATIONS
    1.1 CUT ETCH BETWEEN C1H1 AND GROUND
    1.2 ADD WIRING BETWEEN C1H1 AND C4H1
    1.3 ADD WIRING BETWEEN A1E1 AND C4N2
    1.4 ADD WIRING BETWEEN A1D1 AND C4P2

2.  M7268  MODIFICATIONS
    2.1 COMPONENT SIDE  WIRING
        2.1.1   ADD WIRING BETWEEN E6-4 AND E4-5
        2.1.2   ADD WIRING BETWEEN E5-1 AND E4-4
        2.1.3   ADD WIRING BETWEEN E4-6 AND PIN CH1
    2.2 NON-COMPONENT SIDE  WIRING
        2.2.1   ADD WIRING BETWEEN PIN CN2 AND J1SS
        2.2.2   ADD WIRING BETWEEN PIN CP2 AND J1UU

3.  M7269 MODIFICATIONS
    3.1 ADD A NEW 8881 CALLED E14B
        3.1.1   ADD AN ADDITIONAL DEC 8881 I.C. BY BENDING PINS 1 THROUGH 6
                AND PINS 8 THROUGH 13 INTO A HORIZONTAL PLANE AND MOUNTING
                THIS DEVICE OVER E14 (ALSO A DEC 8881 I.C.) USING DOUBLE SIDED
                TAPE ORIENTING E14B-1 OVER E14-1 AND E14B-7 OVER E14-7 AND
                E14B-14 OVER E14-14 ETC.
        3.1.2   SOLDER E14-7 TO E14B-7
        3.1.3   SOLDER E14-14 TO E14B-14
    3.2 COMPONENT SIDE WIRING
        3.2.1   ADD WIRING BETWEEN R25-3 AND E14B-8 AND E14B-9
        3.2.2   ADD WIRING BETWEEN R25-4 AND E14B-5 AND E14B-6
        3.2.3   ADD WIRING BETWEEN R24-7 AND E14B-2 AND E14B-3 AND E14B-4
        3.2.4   ADD WIRING BETWEEN R24-9 AND E14B-10 AND E14B-11 AND E14B-12
        3.2.5   ADD WIRING BETWEEN E14B-1 AND PIN AD1
        3.2.6   ADD WIRING BETWEEN E14B-13 AND PIN AC1
    3.3 NON-COMPONENT SIDE WIRING
        3.3.1   ADD WIRING BETWEEN J1SS AND R25-4
        3.3.2   ADD WIRING BETWEEN J1UU AND R25-3
```

Dear Dr. DEC:

We have a current program that has a large array (10K words) which is

INTRODUCING VAX/RSTS PROFESSIONAL

(The New and Expanded Version of RSTS PROFESSIONAL)

THE MAGAZINE FOR VAX AND RSTS USERS

Not many things can be called "the one and only."

In 1979, we introduced the one and only independent magazine exclusively for RSTS users. Now, we proudly announce the expansion of that internationally-respected journal into the only magazine for both VAX and RSTS users: VAX/RSTS PROFESSIONAL.

That's right. We expanded RSTS PROFESSIONAL into a new, bigger and more complete magazine created exclusively for both VAX and RSTS users.

We did it, not by cutting down on RSTS articles, but by adding articles on VAX to create a bigger and better journal.

So if you're into VAX, then the new VAX/RSTS PROFESSIONAL was created for you. And if you know anything about RSTS PROFESSIONAL, you'll know the high-quality and technical excellence you can expect.

Our articles are by users whose names you'll recognize; users who know DEC hardware and software because they use it each and every day. Tips, insightful product reviews, answers to your questions, expert opinions and more will be inside every issue — all the things that will give VAX and RSTS users that "professional edge."

Subscribe today to the magazine that is indispensable to every savvy VAX and RSTS user.

NO OTHER MAGAZINE FOCUSES EXCLUSIVELY ON VAX AND RSTS.

Don't miss even one issue. Fill out the postage-paid subscription card and return it today.
Only $35 for six issues.

VAX RSTS PROFESSIONAL

P.O. Box 361, Fort Washington, PA 19034 □ (215) 542-7008

CIRCLE D220 ON READER CARD

initially inputted via a DMA controller (DRV11-B). Initially the processing portion of the program was small, so we located everything in lower 28K (direct addressing).

We will be making additions to this program, so we would like to move the large array into extended memory. There is no problem with using FORTRAN's VIRTUAL array option except for one, the initial input of data from DMA controller; this routine must be written in MACRO (I think).

My basic question: Is there a way to communicate the extended address generated by FORTRAN for VIRTUAL arrays to a MACRO subroutine? We are using the DRV11-B controller and it has provisions for the upper two bits of extended addressing.

Any information would greatly be appreciated.

Rex Klopfenstein, Jr., Vice President
King Industries, Inc.
Bowling Green, OH

There are two answers to your question — an easy answer and a hard one. The easy answer is not to change the existing program at all, but to copy the data from the DR11 into low memory as before. However, after getting the data, write it to a temporary disk file. Then, chain to the program which uses the FORTRAN VIRTUAL arrays and read the data from the temporary disk file into the VIRTUAL arrays. Knowledge of the structure of extended memory is not required if you use this technique.

The hard (and correct) answer is to write the DR11 interface as a standard RT-11 Advanced Programmers Guide. Issue a "read" to the DR11 queue element, and use the $NPPHY routine as documented in Section 7.9.4 of the RT-11 Advanced Programmers Guide. This routine returns the 18- or 20-bit address required by DMA devices.

* * *

Dear Dr. DEC:

Our 2mb. VAX 750 runs two printers, a Data Printer Model 3001 on a MDB controller (LPA0:) and a Dataproducts 2230 on a DEC LP11 controller (LPB0:). The system is lightly loaded, with rarely more than twenty processes, yet LPA0: frequently hesitates for about five seconds at a time while printing. SHOW PROCESS /CONT PRTSYMB1 indicates LEF state with priority 13/8. LPB0: never experiences this situation.

Brian W. Bauske
Datanex, Inc., Eugene, OR

Readers?

* * *

Dear Dr. DEC:

I am running TECO on RT-11, V5.0. The following type of backwards search followed by a match control character is treated by TECO as a

forward search (^S, etc., is entered as "CARET S").

```
-S^N $   OR   -1S^N $   IS TREATED AS   S^N $
-S^S$    OR   -1S^S$    IS TREATED AS   S^S$
NOTE:
 -S $  DOES WORK PROPERLY.    $='ESC' KEY.
```

Dr. Fredrick W. Cotton
Ocean & Atmospheric Science, Inc.
Dobbs Ferry, NY

Any answers?

• • •

In our previous issue we printed a letter from Nick Kushmerick in which he requested some suggestions for sending cursor-positioning escape sequences to his VT-100 terminal for FORTRAN programs. It seems his problem was that when the numbers were sent, leading spaces were also included and it failed. Following are some responses to his call for help. We are all fortunate to have such friends.

From James Simmons (alias 'Captain VAX'), CGR Medical Corporation, Baltimore, MD:

The escape sequences expect to receive the eight bit ASCII character from each individual digit. Leading zeros are acceptable. If one wishes to send a row position of 12 and a column position of 100 then the escape sequence shown in FIGURE B *(page 125)* needs to be sent.

The program example in FIGURE C *(page 140)* uses a 3 character buffer to store row and column values. It will insert leading zeros. This may not be the most efficient way of solving this problem, but it works.

After running the program the screen will look as shown in FIGURE D *(page 140)*.

From Michael B. Morin, Abcor, Inc., Wilmington, MA:

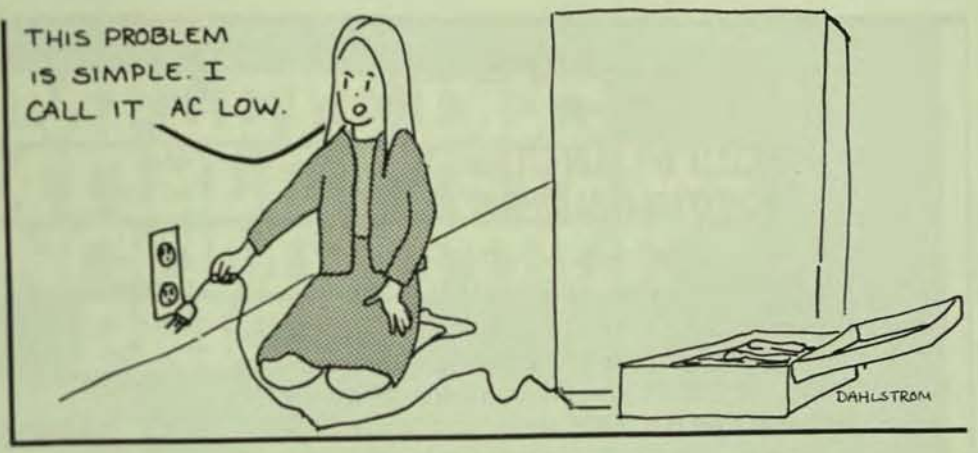
See FIGURE E, page 140.

From Dan H. Cornett, E.I. DuPont de Nemours & Co., Inc., Wilmington, DE:

. . . it is not trivial, but is straightforward. What has to be done is to build up the entire numerical string (with its punctuation), then compress the spaces from the string. The subroutine in FIGURE F *(page 140)* does the trick; except for the use of TYPE (WRITE to default terminal), it follows FORTRAN 77 standards.

An alternative is to use a more general-purpose subroutine which converts any integer to a character string without leading spaces; the use of an "internal write" and subsequent stripping of blank spaces would be

. . . continued on page 125



VT100 GRAPHICS IN A MULTI-LANGUAGE ENVIRONMENT

By Bob Rennert, Director of Academic Computing
Kenyon College, Gambier, OH

Introduction

In many data processing environments it is possible, even desirable, to restrict the programming effort to a primary language. "We're a DIBOL shop," and "We're BASIC-PLUS-2 all the way," are typical phrases I've heard from individuals at DECUS conventions and other gatherings of RSTS users. But in other settings, such as academic computer services, it is often necessary to provide several languages both as a practical requirement (the variety of software obtained from outside sources demands it) and as an appropriate feature of the work supported by the computer system. The very existence of multiple languages, in fact, can be an incentive and invitation for individuals to learn more about computing.

There are, of course, tradeoffs for the computer center staff. The flexibility gained by having many languages may be offset by the potential for fragmentation of programmer expertise. The switching from language to language may also dilute the concentrated attention needed for many projects. Hence for a small programming staff it becomes especially important to develop what Brian Kernighan and P.J. Plauger have called "software tools," a set of common routines which will meet many needs.

At Kenyon College we have recently adapted that concept in the form of library routines which allow users of the several languages we support to take advantage of the many features of the VT100 family of terminals. The paragraphs below describe the content and creation of what we have developed so far.

VT100 Library Routines

The PDP 11/70 RSTS V8.0 system at Kenyon has approximately 40 VT100 or VT101 terminals for student and faculty use. Much of the academic software we use on these terminals was not designed for them specifically and therefore does not take advantage of their potential for selective screen erasure, cursor movement, bidirectional scrolling, reverse video, line drawing and the like. As part of a long-range project to upgrade our software to make it more effective, we developed a common set of routines to handle VT100 escape sequences which we can use from any of the major languages on our system: BASIC-PLUS, BASIC-PLUS-2, FORTRAN, PASCAL (Oregon Software), and C (DECUS distribution).

For each of these languages we have a library located in a logical account called TOOLS:. Each library contains routines which have virtually identical names so that a user could work, for example, with a procedure called CURPOS to position the cursor in any of the languages cited. Listed alphabetically, the routines available are the following:

Routine	Purpose
ANSI	set VT100 for ANSI escape sequences
ATTOFF	turn all graphics attributes off
BLINK	turn on blink mode — VT100 AVO only
BLONWH	set black characters on white screen
BOLD	turn on bold mode — VT100 AVO only
CLRALL	erase all of screen, cursor stays
CLRBOL	erase from beginning of line to cursor
CLREOL	erase from cursor to end of line
CLREOS	erase from cursor to end of screen
CLRHOM	home cursor, clear all screen
CLRLIN	erase all of line
CLRTAB(tab)	clear one or all tab stops
CLRTOC	erase from start of screen to cursor
COL80	set 80 column mode
COL132	set 132 column mode
CURBCK(ncol)	move cursor backward n columns
CURDWN(nrow)	move cursor down n rows
CURFWD(ncol)	move cursor forward no columns
CURHOM	move cursor to home position
CURPOS(row,col)	position cursor at row, column
CURRPT(row,col)	cursor position reported in row, col
CURUP(nrow)	move cursor up n rows
DBLWID	set double width line
DHLBOT	double height line, bottom half
DHLTOP	double height line, top half
G1ASC	load G1 with normal ASCII characters
G1GRAF	load G1 with alternate character set
GRFOFF	load G0 with ASCII character set
GRFON	load G0 with alternate character set
INDEX	active position moves down one line
JUMP	set jump scroll mode
KEYAPP	set keypad application mode
KEYNUM	set keypad to numeric mode
NXTLIN	cursor moves to 1st position on next line
ORGOFF	origin set to upper left of screen
ORGON	origin set to within margins
RESCUR	restore saved cursor attributes
REVIMG	turn reverse image attribute on
REVIND	reverse index, cursor up one line
SAVCUR	save cursor attributes
SETTAB	set tab stop at cursor location
SCREGN(top,bot)	set scrolling region margin between top,bot
SHFIN	select G0 character set
SHFOUT	select G1 character set
SHOLED	turn LED's on/off
SMOOTH	set smooth scrolling mode
SNGWID	set single-width, single-height line
TRMIDN(type)	terminal identification code stored in type
UNDSCR	underscore attribute on
WHONBL	set white characters on black screen

COMPLETE DEC-COMPATIBLE SYSTEMS AT INCOMPATIBLE PRICES

SYSTEMS
(Call for price with J11—LSI—11/73)

System Box*	5"	5"	10"	10"	10"
11/23	—	✓	—	—	—
11/23 PLUS	✓	—	✓	✓	✓
1 Meg Floppys	1	1	1	1	2
Winchester Disk	—	10 Meg	10 Meg	20 Meg	112 Meg
HYPERDRIVE™ RT11 (Solid State Disk)	½ Meg	—	—	—	—
Main Memory	½ Meg	¼ Meg	¼ Meg	1 Meg	1 Meg
Serial Lines	2	—	2	2	10
CRT Terminals	1	—	1	1	8
Tape Backup	—	—	—	—	—
Removable Hard Disk	—	—	—	—	—
	$6,700	$7,000	$8,700	$10,600	$22,000

Trimarchi and Associates can completely configure a system for your particular needs. The above are only a few examples. Systems can be delivered READY TO RUN. We can even SYSGEN the operating system for you. We also offer a large number of DEC and DEC-compatible Q-BUS options at very competitive prices.

OPTIONS

¼ Meg Parity CSR Memory (Dual)	$675
½ Meg Parity CSR Memory (Dual)	1100
1 Meg Parity CSR Memory (Quad)	1575
1 Meg Parity CSR Memory (Quad) with BLOCK MOVE	1750
TTX 3000 Terminal (shown above)	595
LA50 100 CPS Printer	595
LA100 240 CPS Printer	Call
Operating System General License (RT11, RSX11M, etc.)	650
RLV22-AK RL02 with controller (22 Bit)	4250
RL02-AK RL02 Add on drive	2200

Trimarchi and Associates, Inc.
P.O. Box 560
State College, PA 16804
(814) 234-5659

*"JAWS"

DEC is a registered trademark of Digital Equipment Corporation

CIRCLE D135 ON READER CARD

Regular users of VT100 and VT101 terminals will notice the absence of routines to perform certain operations. These were deliberate omissions. Since none of our academic software uses them now and will not in the future, I did not feel a need to code the VT52 mode escape sequences. Other capabilities, such as device status reports, would be used rarely, if ever, by students and faculty and so were left out. And I would prefer that some other sequences not be employed — while I am not positive about the results of running a tight loop for one to a million which issues the code for a powerup-reset on the terminal, I am cautious enough not to try, and think others should be as well.

Some routines take no arguments, and using them is straightforward. Because of the identity in names, the code in main programs is very similar, regardless of the language used. To turn on the reverse image attribute, for example, the programmer would do the following:

Z$ = FNREVIMG$	BASIC-PLUS Function Call
CALL REVIMG	BASIC-PLUS-2 or FORTRAN
REVIMG();	C
REVIMG;	PASCAL

All the routines which take them use integer arguments, either passed explicitly or by means of a variable. For example, to move the cursor forward five columns, the programmer would do the following:

Z$ = FNCURFWD$(5%)	BASIC-PLUS Function Call
CALL CURFWD(5%)	BASIC-PLUS-2
CALL CURFWD(5)	FORTRAN
CURFWD(5);	PASCAL or C

Access to the routines is, of course, language dependent. For BASIC-PLUS, the routines are BASIC-PLUS user-defined functions which can be APPENDed as a set to the user's program. To use them from BASIC-PLUS-2 or FORTRAN, the user would task build or link a compiled main program with an appropriate library: VT1BP2.OLB or VT1FOR.OBJ. For use from PASCAL or C, only slightly more work is involved. In addition to linking a main program module with the right object library, the user must declare the procedures (PASCAL) or functions (C) as "external." For PASCAL, the easiest way is to use a text editor like EDT and pull into the main program a readily available text file (TOOLS:VT1PAS.EXT) which declares all the library procedures, a portion of which looks like this:

```
PROCEDURE ansi;external;        |set VT100 for ANSI escape sequences|
PROCEDURE attoff;external;      |turn all graphics attributes off|
PROCEDURE blink;external;       |turn on blink mode — VT100 AVO only|
PROCEDURE blonwh;external;      |set black chars on white screen|
```

It is then a simple matter of deleting the external declarations which are not needed for that program. Similarly, the user may use an editor to pull into a C program the list of external functions contained in a text file (TOOLS:VTCEXT.H). But since the C compiler has a preprocessor, it is easier to simply include one more line at the top of the program to declare the external functions:

```
#include "tools:vtcext.h"
```

By means of a user manual, we provide information on the routines in several ways. After summarizing the general capabilities of the VT100 family of terminals for cursor movement and alternate graphics sets, we make the typical distinctions between our VT100s (with the Advanced Video Option) and our VT101s: the VT101s cannot do blink or bold, can have only 14 lines of a 132-column screen, and can have reverse image OR underscore but not both at the same time. The routines are then described, in language-independent fashion, grouped by categories such as screen erasure, cursor movement, and line attributes. A separate section discusses the use of the alternate character and line drawing set. This general section is followed by chapters devoted to specific language implementations of the library routines — how to APPEND the BASIC-PLUS user-defined functions and how to call and link the routines to main programs from object libraries in the TOOLS: account. An Appendix provides a set of sample programs, one for each language, which both demonstrate individual procedures as well as show how several procedures can be combined into larger routines.

Common Code for Common Operations

Most of the routines summarized above are relatively easy to implement and involve nothing more than sending the terminal an appropriate escape sequence like "ESC [2J" to clear the screen. While the routines could have been implemented by the equivalent of a PRINT statement in each of the languages, it was decided to use MACRO-11 as the common ground for all except BASIC-PLUS (for which the MACRO-11 routines were converted to user-defined functions).

I do not consider myself an expert MACRO-11 programmer, but sending escape sequences to the terminal from MACRO-11 under RSTS is not very complicated and I have learned much from other professionals. The methods for printing characters to the user's terminal have been documented in pages of this journal (Bob Meyer's "I/O from Macro — Quickly and Easily!," **RSTS PROFESSIONAL**, May/June, 1980, p. 60-61). It was also very helpful to study output routines developed by Brian Nelson at the University of Toledo which can be found on several DECUS RSTS SIG tapes (see, for example, the RSTS SIG tape from Fall, 1980). In fact, the code segments discussed below rely heavily on adaptations of his macros and subroutines for printing characters to the user's terminal.

An example will illustrate. As noted, many procedures take no arguments. The corresponding MACRO-11 subroutine for the "reverse the image" example cited above is the following:

```
revimg::
        .print  #10$,#4
        return

        .nlist  bex
10$:    .ascii  <155.>/[7m/     ;code for reverse video
        .list   bex
        .end
```

The ".print" here is not the macro from the RT11 SYSMAC.SML file but this one (annotated here for purposes of explanation):

MS-DOS® & CP/M® ON A VAX™?*

*Or any other DEC™ system.

YES. You can get all the benefits of personal computing—without buying a personal computer.

Run all the powerful, friendly software packages that have made personal computing so desirable. Virtual Microsystems' total hardware/software solution puts personal computing on *any* DEC system, combining the flexibility and speed of microcomputing with the power and control of your DEC.

For more information on the system that's been proven in hundreds of sites, call (415) 841-9594 or send this coupon to:
Virtual Microsystems
2150 Shattuck Ave, Suite 720
Berkeley, CA 94704.

Tell me more.

Name _____

Company _____

Address _____

City _____

State _____ Zip _____

Telephone ( ) _____

virtual microsystems

DP1-84

Registered trademarks: CP/M: Digital Research; MS-DOS: Microsoft Corp.; VAX: Digital Equipment Corp.
Trademark: DEC: Digital Equipment Corp.

CIRCLE 0211 ON READER CARD

```
.macro   .print  stradr,strlen ;string address, stringlength
   push r5                     ;save R5 for a moment
    .if  b, strlen             ;a string length given?
      .ift
          clr  -(sp)           ;no length, push 0 arg on stack
      .iff
          mov  strlen, -(sp)   ;length given, push on stack
     .endc

   mov stradr,-(sp)            ;push the address argument
   mov sp, r5                  ;move current SP to R5
   callg kbout                 ;global routine to print string

   cmp  (sp)+,(sp)+            ;fix the stack
   pop  r5                     ;restore R5
.endm   .print
```

"Callg" is a macro which sets up a call to a global utility subroutine "KBOUT" that looks like the example on the next page.

KBOUT is a general purpose routine which may be used to print out any null-terminated (e.g. .asciz) string. If the length of the string is not provided, another routine (LEN) calculates it. Since the escape sequence codes needed to do the VT100 routines are short, I always manually count the length and supply it to avoid calling the LEN routine. SAVE and RESTORE invoke macros which push and pop registers onto and from the stack.

The two symbols ".CRSTS" and ".WRITE" deserve more attention. ".WRITE" is the RSTS monitor directive EMT instruction provided in COMMON.MAC and described in the **RSTS System Directives Manual**. The manual also discusses the need for a "prefix" EMT to accompany monitor directives which is dependent on the run-time system being used. Without rehashing that discussion, it is sufficient to note that programs running under run-time systems like RT11 need the prefix EMT while others using RSX do not.

To handle this possibility in a general way, I followed Nelson's lead to create the symbol .CRSTS. Depending on which run-time system will be used, .CRSTS is set equal to either 240 octal, the machine code for NOP (no operation) or EMT 377 (octal), which is the EMT prefix used by RT11 and the Oregon Software PASCAL run-time systems. The definition of .CRSTS is thus conditional, and is controlled at assembly time by another one-line prefix file separate for each language.

To illustrate, consider the case in which I need to assemble the REVIMG routine for both FORTRAN and BP2. In the case of FORTRAN, which uses RT11, I will provide this command line to MACRO.SAV:

REVIMG=KENCOM,FORPRE,REVIMG

For BP2, which uses RSX, the command line to MAC.TSK is

REVIMG=KENCOM,BP2PRE,REVIMG

The KENCOM.MAC file is an abbreviated version of COMMON.MAC which also defines other macros not found in COMMON.MAC. Among these additional macros are the following:

```
.macro $for
   .crsts =     emt + 377
   $for$ = 1
.endm     $for
```

```
.macro $bp2
  .crsts = 240
  $bp2$ = 1
  .endm $bp2
```

The FORPRE prefix file contains one line, a call to the $for macro; likewise the BP2PRE prefix file contains only a call to the macro $bp2. In this way the symbol .CRSTS is defined appropriately for a particular language I wish to use. In similar fashion, I used macros for PASCAL and C in their own prefix files.

```
;           input: (r5) - address of string to print
;                  2(r5) - length of string

kbout::
     save    <r0,r1>         ;save work registers
     mov     2(r5),r0        ;length goes in r0
     bgt     10$             ;if length is given, skip calculation
     mov     (r5),r0         ;put address of string in r0 and
     callg   len             ;find out how long it is

10$: mov     #xrb,r1         ;address of XRB in r1
     mov     r0,(r1)+        ;length goes in xrbc (byte count)
     mov     r0,(r1)+        ;length also goes in xrlen
     mov     (r5),(r1)+      ;address goes in xrloc
     clr     (r1)+           ;channel num *2 (kb = 0)
     clr     (r1)+           ;xrblk not used for kb
     clr     (r1)+           ;xrtime not used for .write
     clr     (r1)+           ;xrmod not used for .write

     .crsta                  ;call rsts
     .write                  ;to do the write

     restore <r1,r0>         ;restore the registers
     return

     .end
```

The ingredients for creating object libraries for these different languages are thus relatively simple: one set of common assembly routines, a set of prefix files which define appropriate symbols, and a set of command files for ATPK which are identical except for the name of the prefix file and the name of the object library I wish to organize. To illustrate, here are a few lines from the command files to generate the VT1FOR.OBJ library and the VT1BP2.OLB library:

```
RUN $MACRO
ANSI=KENCOM.FORPRE.ANSI
ATTOFF=KENCOM.FORPRE.ATTOFF
...
RUN $LIBR
VT1FOR=ANSI/C
ATTOFF/C
etc.

RUN $MAC
ANSI=KENCOM.BP2PRE.ANSI
ATTOFF=KENCOM.BP2PRE.ATTOFF
...
RUN $LBR
VT1BP2/CR:50=ANSI
VT1BP2/IN=ATTOFF
etc.
```

The symbol-defining prefix files are also necessary for routines which take arguments. For example, the move-the-cursor-forward routine, CURFWD, takes an argument for the number of columns. Refer to the code for this routine in Figure 1. The various prefix files, in addition to causing .CRSTS to be defined, also cause symbols like pas and $crt1$ to be defined. In this way it is possible to do a check at assembly time for the language being used and thus to adjust for the way the arguments are passed to the routine. BP2 and FORTRAN use the standard "R5 pointing to a list of arguments" method, Oregon Software PASCAL (version 1.2) pushes arguments on the stack from left to right while DECUS C does the reverse and pushes arguments from right to left.

At assembly time a check is done via a symbol for what language is being used and the proper code is generated for accessing the argument, calling a subroutine common to all the languages, and cleaning up the stack if necessary. At least for our programming staff, this is economical — instead of many routines in many different languages, we have common routines which can be used via conditional assembly blocks.

On a scale of difficulty, the routines which take no arguments are the easiest to implement — simply a matter of printing the appropriate escape sequences to the terminal. The routines which take one or more arguments are slightly more involved and require identifying how the arguments are being passed to the routine and making sure

the routine does any necessary cleanup work before it exits. Once the calling sequence is discovered, though, the conditional assembly blocks have the essentially the same format in each file which requires them.

Two of the routines are more complex and difficult: CURRPT and TRMIDN. They are the only routines which return any information to the calling program. Since they both involve the same methods, I have included the code for only one of them, the CURRPT routine given in Figure 2. The strategy used is to open the user's keyboard with mode 256% (escape sequence mode), then send the escape sequence for reporting on cursor position (or identifying the terminal as the case may be) in binary, and finally parse the escape sequence coming back from the terminal in order to return the proper information to the program. In this way it is possible to prevent the terminal's response from appearing on and cluttering up the display on the screen. Though I am not an expert on RSTS internals, it seems as though the execution of these routines might be dependent on how heavily loaded the system is, how quickly the terminal responds and the like. For that reason I have put in the error traps and error messages; in the documentation on the routines I've asked users to alert me if they get any of them.

LETTERS TO THE DEC PRO...

...from page 6

FORTRAN but nothing on DIBOL. Attempting a normal XCALL simply causes the system to go to lunch. I know this can be done. Can you tell me how or where I can get the appropriate information?

Thanks.

John C. Taylor
Norcross, Georgia

• • •

I enjoyed David Chestnutt's article in the November 1983 issue on "An EDT Adventure." This article was very educational and came at just the right time, as I had started to work on a project incorporating EDT into DIBOL application programs on our VAX/750.

There was a problem I found in using the "Edit Menu" as described on page 106. If, while the menu is displayed, one hits GOLD/M the menu is redisplayed. Then if you hit return, you are returned to EDT but the current buffer is not =Main but is instead =Edit_Menu. The following is the code I put into EDTINI.EDT to fix this and also display Edit_Menu when first getting into EDT.

```
.title  curfwb          ;move cursor forward or backward n columns
.ident /1.0/            ;on VT100, VT101, VK100 (GIGI)

;
;   Bob Bennett
;   Crawford Academic Computer Center
;   Kenyon College
;   Gambier, OH  43022
;
;   !32 used as max for column move since users may expect it (!31
;   is real limit)
;
        .enabl lsb

curfwd::
        movb    #^C,@2$         ;variable code for forward
        movb    #^C,1!1$
        br      5$

curbck::
        movb    #^D,@2$         ;variable code for backward
        movb    #^D,1!1$

5$:     .if     df $pas$        ;running under pascal?
        tst     2(sp)           ;yes, got a value for columns > 07
        ble     3$              ;columns <= 0
1$:     cmp     2(sp),#132.     ;upper limit for column move
        bgt     3$              ;out of bounds
        push    r0              ;save a register
        mov     4(sp),r0        ;parameter goes here
        call    100$            ;do the subroutine
        pop     r0              ;restore r0
        br      10$             ;all done
3$:     .print  #110$,#4        ;columns out of bounds, default to 1 column
10$:    mov     (sp),2(sp)      ;adjust stack for pascal
        tst     (sp)+
        return                  ;and get out
        .endc

        .if     df $crt1$!$cras$ ;running from c?
        tst     2(sp)           ;got a value for columns > 07
        ble     13$             ;columns <= 0
```

```
I = Edit_Menu


---------- EDIT MENU ------------

RETURN  =  RETURN TO THE DOCUMENT

GOLD/F  =  FILE THE DOCUMENT WITH CHANGES

GOLD/Q  =  QUIT, NO CHANGES TO DOCUMENT

GOLD/M  =  DISPLAY THIS MENU


    ^I
    DEF M SHO_MENU
    I =SHO_MENU
    DEF K CON M AS "EXT RET-MENU."
    DEF K G M AS "EXT RET_MENU."
    FIND =EDIT_MENU 20
    ^I
    DEF M SHO_MENU_1
    I =SHO_MENU_1
    DEF K CON M AS "EXT RET_MENU_1."
    DEF K G M AS "EXT RET_MENU_1."
    FIND =EDIT_MENU 20
    ^I
    DEF M RET_MENU
    I =RET_MENU
    DEF K CON M AS "^M."
    DEF K G M AS "EXT SHO_MENU."
    F L
    ^I
    DEF M RET_MENU_1
    I =RET_MENU_1
    DEF K CON M AS "^M."
    DEF K G M AS "EXT SHO_MENU."
    F =MAIN
    ^I
    DEF K G M AS "EXT SHO_MENU."
    DEF M FINEXIT
    I =FINEXIT
    EXIT
    ^I
    DEF M FINQUIT
    I =FINQUIT
    QUIT
    ^I
    DEF K G AS "EXT FINEXIT."
    DEF K G Q AS "EXT FINQUIT."
    SET TERMINAL VT100
    SET MODE CHANGE
    SE WR 78
    SE SC 80
    SHO_MENU_1
```

I hope to see more articles on new uses for EDT in future editions of *DEC PRO*. Keep up the good work.

Richard A. Kandetzki
Senior Systems Analyst
Georgia-Pacific Corporation
Darien, CT

```
11$:    cmp     2(sp),#132.     ;upper limit for column move
        bgt     13$             ;out of bounds
        push    r0              ;save a register
        mov     4(sp),r0        ;parameter goes here
        call    100$            ;do the subroutine
        pop     r0              ;restore r0
        br      15$             ;all done
13$:    .print  #110$,#4        ;columns out of bounds, default to 1 column

15$:    return                  ;all done
        .endc                   ;conditional code by Sharryn Cory
        .if     df $for$!$bp2$  ;running from fortran or bp2?
        tst     #2(r5)          ;value for columns > 0?
        ble     7$              ;columns < 0
        cmp     #2(r5),#132     ;132 is upper limit
        bgt     7$              ;too big
        save    r0              ;save the register
        mov     #2(r5),r0       ;move parameter to r0
        call    100$            ;do the subroutine
        restore r0              ;restore the register
        br      20$             ;skip over next line
7$:     .print  #110$,#4        ;columns out of bounds, default to 1 column
20$:    return                  ;all done
        .endc

100$:
        .itoa3  r0,#81$         ;convert the integer to 3 ascii characters
        .print  #80$,#6         ;and print the code to move column
        return

        .nlist  bex

80$:    .byte   155.,'[
81$:    .blkb   3
82$:    .byte   0

110$:   .ascii  <155.>/[1/      ;move 1 column
111$:   .byte   0
        .even

        .list   bex
        .dsabl  lsb
        .end
```

```
        .title  currpt          ;get active cursor position from VT100,VT101
        .ident  /1.0/           ;VK100 (GIGI)

;       Bob Rennert
;       Director of Academic Computing
;       Crawford Academic Computer Center
;       Kenyon College
;       Gambier, OH  43022

;       at common routine 100$:
;       input -
;       output - r0 has integer value of row or -1 if error encountered
;                r1 has integer value of column or clear if error encountered
;
;       channel #11 used for kb i/o


currpt::
        .if     df      $pas$   ;running from pascal?
        save    <r0,r1>         ;save a couple of registers
        call    100$            ;do the common routine
        mov     r0,@10(sp)      ;row gets returned here
        mov     r1,@6(sp)       ;column gets returned here
        restore <r1,r0>         ;unsave work registers
        mov     (sp),4(sp)      ;fix stack for pascal
        cmp     (sp)+,(sp)+
        return
        .endc

        .if     df $ort1$!$crsx$ ;running from o?
        save    <r0,r1>         ;save a couple of registers
        call    100$            ;do the common routine
        mov     r0,@6(sp)       ;row gets returned here
        mov     r1,@10(sp)      ;column gets returned here
        restore <r1,r0>         ;unsave work registers
        return
        .endc

                                ;conditional code by Sharryn Cory
        .if df $for$!$bp2$      ;running from any of these?
        save    <r0,r1>         ;save a couple of registers
        call    100$            ;do the common routine
        mov     r0,@2(r5)       ;row gets returned here
        mov     r1,@4(r5)       ;column gets returned here
        restore <r1,r0>         ;unsave work registers
        return
        .endc

100$:
        call$   zfirqb              ;zero the FIRQB
        movb    #2,@#firqb+fqfun    ;OPNFQ function - open file
        movb    #11*2,@#firqb+fqfil ;channel #11
        mov     #"KB,@#firqb+fqdev  ;KB is what we want to open
        mov     #100400,@#firqb+fqmode ;mode 256 (bit 15 must be on)
        .crsts                      ;call rsts monitor
        calfip                      ;get FIP to do it
        tstb    @#firqb             ;get an error?
        beq     3$                  ;no
        .print  #62$,#34.           ;yep, broadcast message
        mov     #-1,r0              ;flag error with a -1
        return                      ;and get out

3$:
        call$   zxrb                ;zero the XRB first
        mov     #4,@#xrb+xrlen      ;4 bytes to write
        mov     #4,@#xrb+xrbc       ;ditto
        movb    #2*11,@#xrb+xrci    ;channel 11
        mov     #60$,@#xrb+xrloc    ;location of the code to write
        mov     #4096.,@#xrb+xrmod  ;PUT channel 11, RECORD 4096 (binary)
        .crsts                      ;call rsts
        .write                      ;write it out there
        tstb    @#firqb             ;get an error on the write?
        beq     5$                  ;no
        .print  #64$,#34.           ;broadcast error message
        mov     #-1,r0              ;flag error here
        call    170$                ;close the kb
        return                      ;and get out

5$:
        mov     #10.,@#xrb+xrlen    ;size of buffer goes here
        mov     #80$,@#xrb+xrloc    ;location of buffer here
        movb    #2*11,@#xrb+xrci    ;channel #11
        mov     #2,@#xrb+xrtime     ;give it a 2 second wait time
        .crsts                      ;call rsts monitor
        .read                       ;read anything from terminal

        tstb    @#firqb             ;get error on read?
        beq     20$                 ;no

15$:    .print  #63$,#34.           ;broadcast error message
        mov     #-1,r0              ;-1 flag for the error to return
        call    170$                ;close kb
        return                      ;and get out


20$:
        call    170$                ;close the kb - don't need it now
        save    <r2,r3,r4>          ;more work registers
        mov     #80$,r1             ;address of buffer here
        clr     r3                  ;use r3 as accumulator,r2 as scratch
        clr     r4                  ;r4 is counter

22$:    cmp     r4,#10.             ;see if we've parsed this much
        ble     23$                 ;ok
        jmp     90$                 ;we've got an error
23$:    cmpb    (r1)+,#'[           ;find the left square bracket
        beq     25$                 ;got it
        inc     r4                  ;bump counter
        br      22$                 ;and check again

25$:    movb    (r1)+,r2            ;current ascii digit in r2
        mul     #10.,r3             ;multiply contents of r3 by 10
        sub     #60,r2              ;convert ascii digit in r2 to integer value
        add     r2,r3               ;add it to running total in r3
        cmpb    (r1),#';            ;got the semicolon separating coordinates?
        bne     25$                 ;no, do another digit
        mov     r3,r0               ;store value of integer row here for return
        inc     r1                  ;push pointer past semicolon
        clr     r3                  ;zero the accumulator
30$:    movb    (r1)+,r2            ;ascii digit into r2 again
        mul     #10.,r3             ;accumulator contents x 10
        sub     #60,r2              ;convert ascii digit to integer
        add     r2,r3               ;add to running total
        cmpb    (r1),#'R            ;end of VT's cursor report?
        bne     30$                 ;not yet
        mov     r3,r1               ;put column value here for return
        restore <r4,r3,r2>          ;unsave work registers
        return                      ;all done

170$:   call$   zfirqb              ;zap firqb again
        movb    #clsfq,@#firqb+fqfun ;close channel code
        movb    #2*11,@#firqb+fqfil ;channel number here
        .crsts
        calfip                      ;FIP to close channel
        tstb    @#firqb             ;error on the close?
        beq     171$                ;no,
        .print  #65$,#34.           ;signal error
        mov     #-1,r0              ;flag error for return
171$:
        return                      ;all done

60$:    .ascii  <155.>/[6n/     ;cursor report code
62$:    .asciz  <7>/?Error in opening KB from CURRPT /
63$:    .asciz  <7>/?Error in reading KB from CURRPT /
64$:    .asciz  <7>/?Error writing to KB from CURRPT /
65$:    .asciz  <7>/?Error closing KB in CURRPT      /
66$:    .asciz  <7>/?Error parsing report in CURRPT  /
        .even

80$:    .blkb   10.                 ;buffer for kb read

90$:    .print  #66$,#34.           ;error on cursor report
        mov     -1,r0               ;signal error
        call    170$                ;close the kb
        return                      ;and get out

        .end
```

Summary

In a multi-language environment, it is possible to have a common set of routines to access the graphics capabilities of the VT100 family of terminals. By means of conditional assembly blocks and prefix files, object libraries can be created which can be linked with several languages on RSTS systems. The standardization of "tools" thus achieved can help simplify and systematize the programming tasks for many projects.

FIGURE B

```
                    <ESC>  [    1    2    ;    1    0    0    H
HEX values            1B   5B   31   32   3B   31   30   30   48

ALSO ACCEPABLE IS

                    <ESC>  [    0    1    2    ;    1    0    0    H
HEX values            1B   5B   30   31   32   3B   31   30   30   48
```

. . . continued from page 115

done in the same manner as used by this routine.

P.S. For multi-terminal portability, any existing library cursor routine should be used, such as are available on VAX/VMS.

From Harry Flowers, Univ. of Tenn., Center for the Health Sciences, Memphis, TN:

I have included two libraries of subroutines for screen formatting. VT52.FOR [*FIGURE G, page 142*] contains routines for a VT52 or using a VT100 in VT52 mode. ANSI.FOR [*FIGURE H, page 142 and 144*] contains routines for a VT100. The direct cursor addressing is not as efficient in ANSI mode, so unless you need some of the other ANSI features, I suggest using the VT52 subroutines. A program, SCREEN.FOR [*FIGURE I, page 144*], is provided to demonstrate the ANSI group of subroutines. It is easy to see how to write other subroutines for screen control from those given. Some (WIDE, LARGE, SINGLE) require the advanced video option on a VT100, or a VT101 or VT102. If you don't have any of these types of terminals, then you might as well use the VT52 subroutines.

Alternately, a call to subroutine VT52 at the beginning of DCA (the VT52 one) and a call to ANSI at the end would accomplish the same as the ANSI subroutine DCA if the terminal has VT52 compatibility. This will work on a VT100.

NOTICE: The author and the University of Tennessee assume no responsibility for the use or reliability of this software. This software cannot be sold for profit, but may be copied and used otherwise.

* * *

. . . continued on page 140

Once on a world

long ago... you typed and retyped letters, documents and manuals... filed and often retrieved documents... eventually hand-crafted nifty spreadsheets... made charts... scribbled on your calendar... and got writer's cramp...

TRANSFERRING FILES BETWEEN RSTS AND UNIX

By James B. Wilkinson, Associate Professor, Computer Science Dept., College of Charleston, Charleston, SC

The problem of moving files between operating systems is not an uncommon one, and it arises in many forms. We were interested in moving files between RSTS and UNIX because those are the two operating systems we have. Since we have only one computer, we are forced to use an external medium to hold the data to be transferred while the operating system is being changed. Magnetic tape is a convenient choice, but the two operating systems differ greatly in the way they deal with this medium. This is particularly true of the way in which directory structures are implemented on tape. It seems very difficult to teach either operating system to handle the magtape directory structure of the other. Tapes with no directory structure are another possibility, and this article describes a simple method we have found to use such tapes for the purpose.

The method requires two small BASIC-PLUS programs for RSTS/E and three shell scripts for UNIX. We adopted 512bytes as the size for our tape records since both operating systems use that size for disk blocks.

THE NULL-FILL PROBLEM

RSTS pads a file with nulls to the end of the last disk block. UNIX keeps in the directory a count of the actual number of characters in the file, while RSTS does not — at least in the general case. When a file is transferred from RSTS to UNIX, UNIX will believe that the nulls are actually part of the file. This caused us problems when the file transferred was an include file for a C program. Since the padding nulls follow the last new line in the file they will appear to the compiler to be at the beginning of the next line of source code following the line that included the transferred file. If that line is also an instruction to the preprocessor, these nulls at the beginning will keep the '#' from being first on the line; this means the preprocessor will not process the instruction.

This problem can be fixed by piping the transferred file through the command 'tr' which trims off the nulls. Since 'tr' deletes all nulls, and since nulls in binary files can be significant, these files are not treated in this way.

The complementary problem in taking files from UNIX to RSTS is to add null padding to all files.

THE CARRIAGE-RETURN PROBLEM

Most text files on RSTS use the sequence '<carriage return> <line feed>' to separate lines, while UNIX files use only the <line feed>. This means that files brought to RSTS from UNIX need to have a <carriage return> inserted before every <line feed> before they are examined or edited. The program CR does this.

Text files taken from RSTS to UNIX should have the <carriage return> deleted. This is also done with 'tr'. Note carefully that this will take out all the <carriage returns>. Although one can imagine circumstances under which this is not what is wanted, we haven't found it worthwhile to do anything about it. Forewarned is forearmed.

Of course, none of the above applies to binary files.

RSTS PROGRAMS

XUNIX

This program moves the data between the magtape and the RSTS disks. It asks 'Which way magtape (in or out)?'. This question should be answered with 'I' if the tape is the input medium (i.e., files will be read from the tape) or with 'O' if the tape is the output medium. The program then goes into a loop which asks for the name of the disk file to be written or read, does the transfer after the name is typed, and reports the number of blocks transferred. A Control/Z will cause the program to exit. The program also exits after a zero-block transfer from an input magtape (i.e., the end-of-tape marker is reached).

CR

This program puts a <carriage return> before each <line feed> in a file. It is used to process text files read into the RSTS system. It repeatedly asks for a file name and processes the file whose name is given in response. Again, a Control/Z will break the loop.

ATPK Command Files

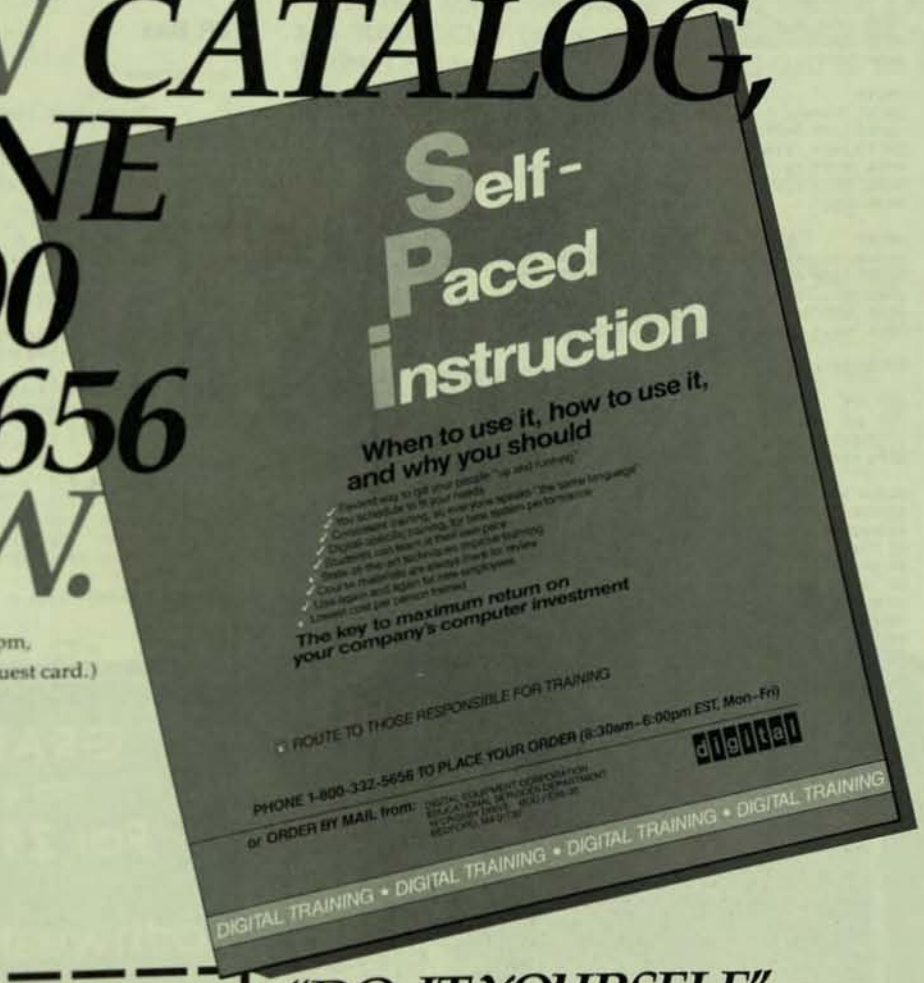
It is easy to edit a file containing the names of the files to be transferred so that it makes a suitable command file for ATPK. All that is required is to put 'RUN XUNIX' followed by an 'I' or 'O' at the beginning and a ↑Z at the end. The names of input text files should be duplicated so the files can be passed through CR as well (see the sample files).

UNIX PROGRAMS

These programs are all shell scripts that invoke the command 'dd' to do the actual transfers. Two versions of the input program are provided : 'fmrsts_text' for reading text files from tape, and 'fmrsts_bin' for reading binary files. Only one, namely 'torsts', is needed to write tape files. Each of the script files should be passed through the '+x' option of 'chmod' before it is used. One of these programs is run by typing its name followed by a list of file names. If the names are available in a file, then the command substitution mechanism of the shell may be used as demonstrated in the procedures.

IF YOU DON'T HAVE THIS NEW CATALOG, PHONE 1-800 332-5656 NOW.

(between 8:30 am and 6 pm,
Monday thru Friday, or send request card.)



☐ YES. SEND ME the new Digital Mail Order catalog on Self-Paced Instruction courses.

My Name _____

Company _____

Department _____ Title _____

Mailing Address _____

_____ Zip _____

☐ SEND ONE TO OUR TRAINING DIRECTOR, TOO.

Name _____

Company _____

Mailing Address _____

_____ Zip _____

USE THIS FORM TO REQUEST SPI CATALOG BY MAIL.

digital Digital Equipment Corporation
Educational Services Department
12 Crosby Drive BUO/E55-20
Bedford, MA 01730

"DO-IT-YOURSELF" TRAINING THAT PAYS... FROM DIGITAL

If you want to be 'up and running' fast, on *any* Digital system, then SPI training is your answer. It is the lowest-cost way to train anyone, and it produces learning that sticks, skills that expand.

This comprehensive new catalog tells you which SPI courses you should consider, exactly what is available and how to order.

If you are a Systems Analyst, Programmer, Operator, involved in maintenance, or simply a User, you should look into this catalog . . . and so should your Training Director.

Ask for your copy today.

Digital Equipment Corporation
Educational Services Department
12 Crosby Drive BUO/E55-20
Bedford, MA 01730

digital

CIRCLE D103 ON READER CARD

LISTINGS

XUNIX.BAS

```
1000 EXTEND
!
!       I/O for non-file structured mag-tapes
!
!       21-Mar-83       J. Wilkinson    College of Charleston

1010    ON ERROR GOTO 1090

1020    PRINT
\       INPUT "Which way magtape (in or out)"; DIRECTION$
\       DIRECTION$ = LEFT(CVT$$(DIRECTION$, 38%), 1%)

1030    GOTO 1040 IF DIRECTION$ = "I"
\       GOTO 1050 IF DIRECTION$ = "O"
\       GOTO 1020                                       ! Make it be right

1040    PRINT
\       PRINT "Output file name";
\       INPUT LINE OUTFILE$
\       OUTFILE$ = CVT$$(OUTFILE$, 38%)
\       OPEN OUTFILE$ FOR OUTPUT AS FILE 1%, MODE 32%
\       OPEN "MT:" FOR INPUT AS FILE 2%
\       GOTO 1060

1050    PRINT
\       PRINT "Input file name";
\       INPUT LINE INFILE$
\       INFILE$ = CVT$$(INFILE$, 38%)
\       OPEN INFILE$ FOR INPUT AS FILE 2%
\       OPEN "MT:" FOR OUTPUT AS FILE 1%

1060    COUNTER% = 0%

1070    GET #2%                                         ! Main loop
1080    PUT #1% + SWAP%(2%)
\       COUNTER% = COUNTER% + 1%
\       GOTO 1070

1090    CLOSE 2%
\       RESUME 1040 IF ERR = 5% AND ERL = 1040%
\       RESUME 1050 IF ERR = 5% AND ERL = 1050%
\       ON ERROR GOTO 0 IF ERR <> 11%
\       RESUME 32767    IF ERL = 1020% OR ERL = 1040% OR ERL = 1050%
\       ON ERROR GOTO 0 IF ERL <> 1070%
\       RESUME 1100     IF COUNTER% = 0%
\       PRINT COUNTER% ; "block";
\       PRINT "s"; IF COUNTER% <> 1%
\       PRINT " read from tape." IF DIRECTION$ = "I"
\       PRINT " written to tape." IF DIRECTION$ = "O"
\       CLOSE 1%
\       RESUME 1030

1100    IF DIRECTION$ = "I"
           THEN
                PRINT "No more files."
\               CLOSE -1%

1110    IF DIRECTION$ = "O"
           THEN
                PRINT "Empty input file not written on tape."
\               GOTO 1030
32767   END
```

CR.BAS

```
1000 EXTEND
!
!       Change each line to <text> <CR> <LF>
!
!       21-Mar-83       F. Mitchell     Erskine College

1010    ON ERROR GOTO 8000

1020    PRINT
\       PRINT "Input file";
\       INPUT LINE IN.FILE$
\       IN.FILE$ = CVT$$( IN.FILE$, 38% )
\       GOTO 1020 IF LEN( IN.FILE$ ) = 0%

1040    OPEN IN.FILE$ FOR INPUT AS FILE 1%
\       OPEN IN.FILE$ FOR OUTPUT AS FILE 2%, MODE 32%

2000    INPUT LINE #1%, T$
\       PRINT #2%, CVT$$( T$, 4% )
\       GOTO 2000

3000    CLOSE 1%, 2%
\       GOTO 1020

8000    RESUME  3000 IF ERR = 11% AND ERL = 2000%
\       RESUME  1020 IF ERR =  5% AND ERL = 1040%
\       RESUME 32767 IF ERR = 11% AND ERL = 1020%
\       CLOSE 1%, -2%
\       ON ERROR GOTO 0

32767   END
```

Fmrsts_text

```
for i
do
echo $i
dd if=/dev/nrmt0 bs=1b | tr -d '\015' > $i
done
```

Fmrsts—bin

```
for i
do
echo $i
dd if=/dev/nrmt0 of=$i bs=1b
done
```

Torsts

```
for i
do
echo $i
dd if=$i of=/dev/nrmt0 bs=1b
  conv=sync
done
```

RSTS TO UNIX PROCEDURE

1. Get all the files to be transferred into the account where you are logged in. Put the names of the text files and the binary files into two new text files called 'TNAMES.TXT' and 'BNAMES.TXT'. Use your editor to create an ATPK command file which is the concatenation of the two name files. Insert the name of each name file just before its contents. At the beginning insert the lines 'RUN XUNIX' and '0'. At the end insert the line '↓Z'.

2. Mount a reel of tape on drive 0.

3. Give the command '@ COMAND'.

4. After the tape is written mount it on drive 0 of the UNIX system.

5. Type the following sequence of commands:

'fmrsts text tnames.txt'
'fmrsts text 'cat tnames.txt''
'fmrsts_text bnames.txt'
'fmrsts_bin 'cat bnames.txt''

UNIX TO RSTS PROCEDURE

1. Get all the files to be transferred into the directory where you are logged in. Create a file called 'names' consisting of the names of the text files followed by the names of the binary files. Put the name of the names file at the beginning.

2. Mount a reel of tape on drive 0.

3. Type 'torsts 'cat names''.

4. After all the files have been written mount the tape on drive 0 of the RSTS system.

5. Run XUNIX, and answer 'I' to the direction question and 'NAMES.CMD' to the output file question. Then type a Control/Z. Run CR and answer 'NAMES.CMD' to the input file question. Then type a Control/Z.

6. Edit 'NAMES.CMD' as follows:
   1. Delete the name of the names file (the first line).
   2. Add the lines 'RUN XUNIX' and 'I' at the beginning.
   3. Add the lines '↑Z', 'RUN CR', '↑Z' at the end.
   4. Insert a copy of the names of the text files immediately after the 'RUN CR'.
7. Give the command '@ NAMES'.

SAMPLE FILES

In the headers of this section the names of RSTS files are in upper case, and the names of UNIX files are in lower case. This respects ordinary usage and makes it easier to tell which is which. Some care must be taken with cases in UNIX since that operating system considers upper and lower cases to be separate characters. RSTS does not distinguish between cases.

RSTS to UNIX

TNAMES.TXT —
one.txt
two.txt
three.txt

BNAMES.TXT —
one.bin
two.bin

COMAND.CMD —
RUN XUNIX
0
tnames.txt
one.txt
two.txt
three.txt
bnames.txt
one.bin
two.bin
↑Z

UNIX to RSTS

Names —
names
four.txt
five.txt
four.bin
five.bin
six.bin

NAMES.CMD —
RUN XUNIX
I
four.txt
five.txt
four.bin
five.bin
six.bin
↑Z
RUN CR
four.txt
five.txt
↑Z

ACKNOWLEDGEMENTS

Frank Mitchell of Erskine College wrote CR and fixed and beautified XUNIX. Billy Glidden of the Collge of Charleston and Ken Sallenger and Steve Wright of the University of South Carolina helped check out the programs.

UNIX is a trademark of Bell Telephone Laboratories, Inc.
RSTS/E is a trademark of Digital Equipment Corp.

DEC PROFESSIONAL CROSSWORD

By John K. Young

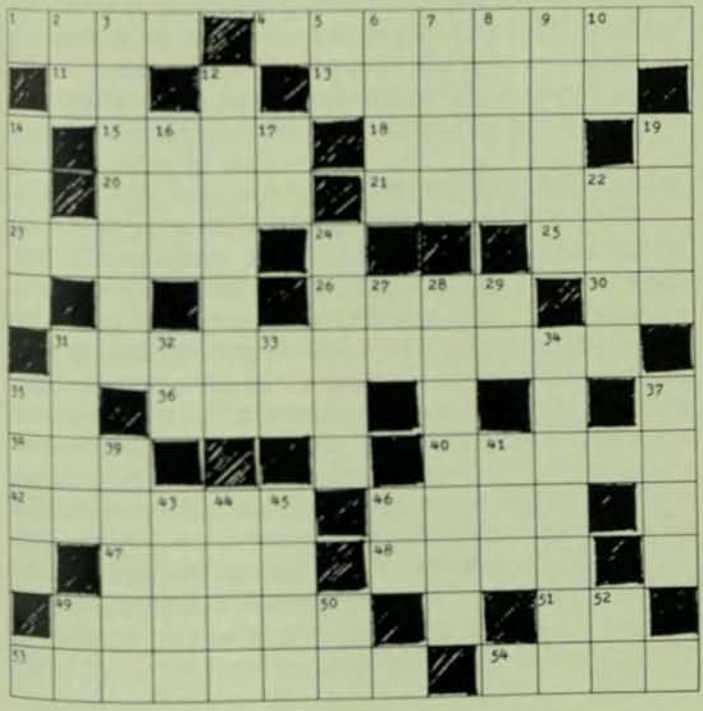
(Answer next issue.)

DOWN

2 At the time that
3 Gather together in proper order
5 Written above (Abb.)
6 Be scattered over
7 Sea eagle
8 Some of it will be replaced by base terminals
9 Characters of eight bits together
10 Place of the seal (Abb.)
12 Instructions telling computer what to do
14 It stores data
16 Electrically charged group of atoms
17 This news syndicate, like the others, depends on computers (Abb.)
19 Storage for information unit
22 Kind of work the computer can reduce dramatically
24 Set of columns in punch cards
27 Tar Heel State (Abb.)
28 Computer that uses numbers
29 Word of affirmation in mantra
31 Saudi Arabia region on Persian Gulf
32 Xenon (Chem.)
33 Prefix meaning "double"
34 Specific location in computer's memory for stored information
35 Transfer to another section of storage
37 Circuit with two inputs and one output
39 Binary code for numbers, letters, and other symbols
41 Short, light sleep
43 Divided with uncertainty
44 Prepare for presentation by cutting and slicing
45 Divorce capital
46 Public road (Abb.)
49 100 square meters
50 Genitourinary (Abb.)
52 That fellow

ACROSS

1 Combine different fields of info into one machine word
4 Kind of language computers understand
11 In precisely this manner
13 Static interfering with radio reception
15 Italian monetary unit
18 Smallest whole number
20 Repeated sequence of operations
21 Eliminate from computer's memory
23 Specialized vocabulary of those in computer work
25 Roman sun god
26 Combining form meaning "of Hindu stock"
30 Suffix meaning "having to do with"
31 16-digit number system
35 Russian "yes"
36 Man's name
38 Country that leads in computer technology
40 It's headed by Mrs. Ghandi
42 Program that sets a standard
46 *
47 This will be done better if programmer knows it will be read
48 Form for pre-recorded program
49 Broadcasting
51 Be quiet!
53 Computer's output in typewritten form
54 Second-hand



SELECTING THE PROPER COMMUNICATIONS MULTIPLEXERS FOR YOUR DEC COMPUTER SYSTEM

By Norm Kiefer, Emulex Corporation

When interfacing terminals to your DEC processor, it is important to understand the operational characteristics of the DEC DZ11, DH11, and DMF32 communications multiplexers available in today's market. Since multiplexer selection can have a significant impact on system throughput, proper selection is a critical consideration when configuring your system. Incorrect application of the multiplexer can cause additional interrupt overhead on the CPU, substantially reducing the overall performance of your system. Since selection of the proper multiplexer also involves knowledge of the applications running on your system, an attempt will be made here to relate the performance aspects of the different multiplexers to the application areas where each would be most efficient.

INPUT CHARACTERISTICS

The DEC DZ11, DH11 and DMF32 multiplexers use similar methods for handling input characters. Each of these devices employs input silos for temporary storage of incoming characters. An Interrupt is generated to the CPU when a character is present in the input silo. The processor must then respond to the Interrupt by executing the proper software routine in the device driver to fetch the character from the input silo. The device driver examines the character, stores it in memory, and echos it back to the terminal for display to the operator. If the input character is a control character or control sequence, the CPU then executes the appropriate software routine. This type of operation is referred to as Programmed Input/Output (PIO) because the CPU is executing instructions to transfer the characters in or out of memory.

PIO INPUT

For input operations, Programmed I/O was selected because it is the most efficient method of handling single character transfers or small numbers of characters. We believe the DEC DZ11, DH11 and DMF32 were originally designed under the presumption that the input device would be a dumb (unbuffered) terminal with input data being limited to the speed of a human typist. Since human reactions are extremely slow by computer standards, even with multiple terminals interfaced to the multiplexers, this type of input is most often processed a single character at a time.

PIO LIMITATIONS

Unfortunately the above assumptions are not always true in the real world. When users interface intelligent terminals, which have internal buffering and can transmit 1024 characters or more in a single burst at 9600 baud, processing data a single character at a time becomes much less efficient. The higher data rates involved in these applications require the computer to spend considerably more time in the device driver, leaving less time to run the user's application software.

BUFFERED PIO OPERATIONS

There are several ways to increase the efficiency of a PIO operation. DEC has structured the device driver in such a way that if there are multiple characters in the input silo of the multiplexer, the processor will transfer the characters in a single burst until the silo is empty. Since multiple characters are transferred with only a single Interrupt to the CPU, this method is more efficient than having the multiplexer generate an Interrupt for each input character. This can only occur when input conditions are such that multiple characters are allowed to accumulate in the silo before the CPU responds to an Interrupt. In order to increase efficiency of the DMF32, DEC has provided a programmable silo interrupt delay. The silo delay provides a user programmable delay between the time the first character is stored in the input silo and the time an Interrupt is generated. For example, a delay of approximately 30 MS is brief enough to be transparent to the user, who will not see the character displayed on his CRT until it is processed by the I/O driver. However, under heavy input conditions, the delay will increase the possibility of multiple characters accumulating in the silo before the Interrupt is sent to the CPU. Although Buffered Programmed I/O increases the efficiency of character input, none of these multiplexers efficiently handles block mode data input from buffered terminals. They were designed for interfacing unbuffered terminals whose data speeds are limited to human response time.

OUTPUT CHARACTERISTICS

We can identify the **input** similarities of the DZ11, DH11 and DMF32, and see the reason for those similarities. The **output** characteristics of each of these multiplexers, however, are distinctly different. These characteristics, when considered with the user application, determine which device will be most efficient.

DZ11

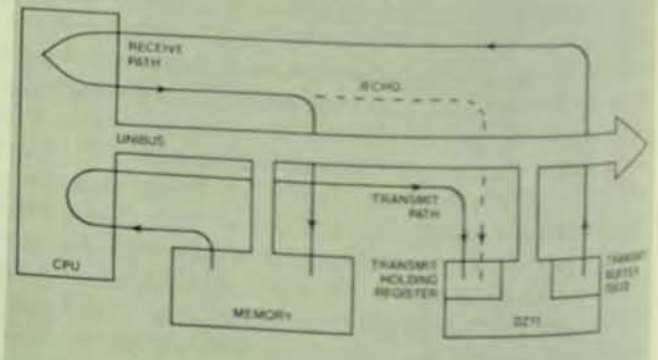
The output characteristics of the DZ11 multiplexer are the easiest to describe since it utilizes the Programmed I/O technique to input and output characters. There is a significant difference between the buffered input of characters and the non-buffered output. The DZ11 handles output a single character at a time. After each character has been transmitted to, and processed by the DZ11, an interrupt is generated as an indication to the CPU to provide the next character to be transmitted. An application well suited to the DZ11 is data entry, where the majority of input and output operations are single character and efficient character echo is important.

Conversely, the DZ11 is not efficient when the application requires the computer to output large blocks of characters. Since the DZ11 requires the CPU to enter the device driver for each output character, substantial amounts of CPU time are used in transferring characters, rather than running application software. The DZ11 is also not the best choice for inquiry applications, where the terminal operator requests substantial amounts of information; for word processing, which frequently sends out large blocks of characters to reformat the CRT screen; or for graphics applications.

DH11

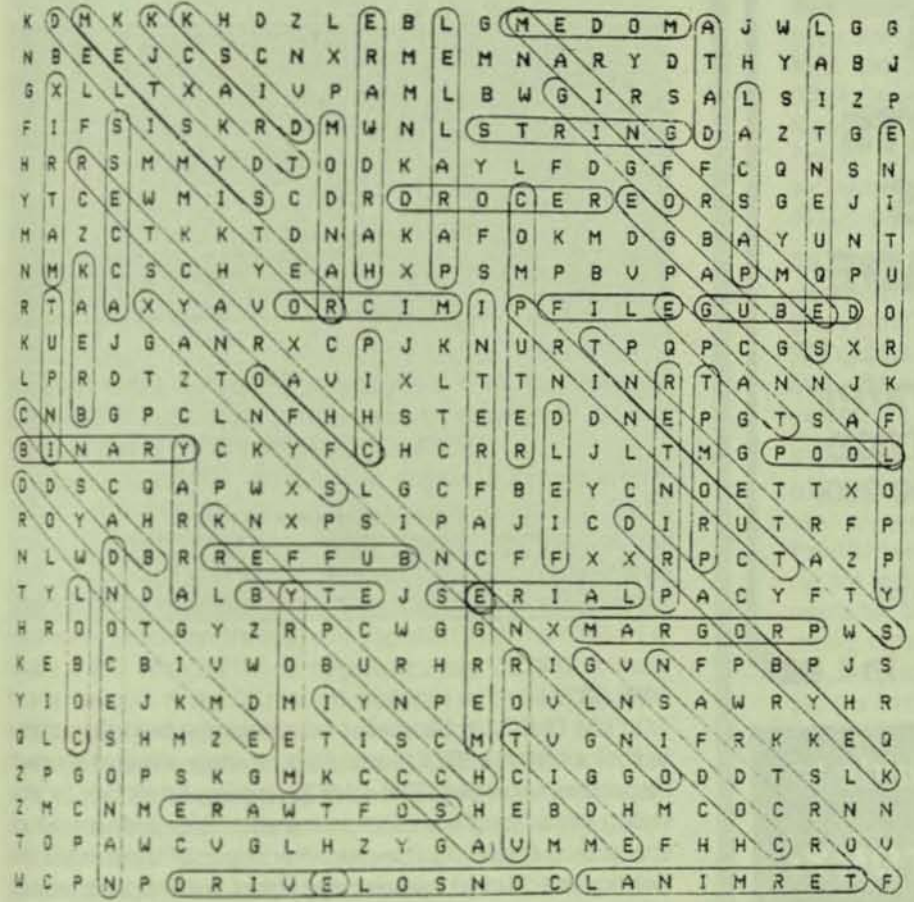
The DH11 multiplexer addresses and solves many of the problems associated with outputting large blocks or characters in inquiry, word processing, and graphics applications. The DH11 reduces CPU overhead by using Direct Memory Access (DMA) output. DMA is a process where the CPU tells the controller the memory starting address and the character count of the data to be transferred. The controller can then utilize the computer bus to transfer characters from main memory to the terminal, while the CPU is occupied with internal computing functions. Other than providing DMA "set-up information" to the controller, the CPU does

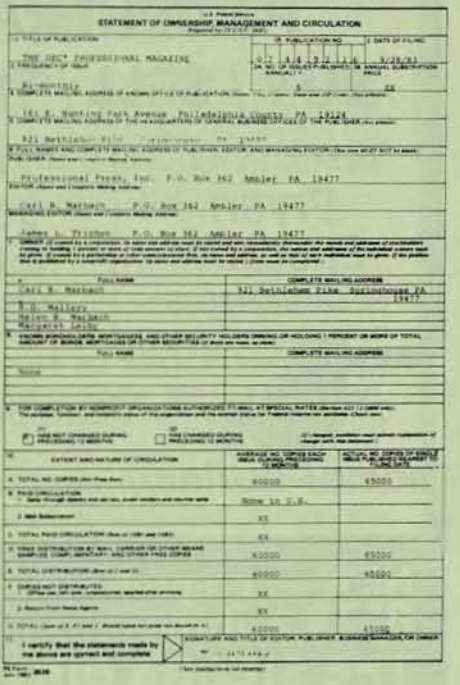
FIGURE 1



DZ11 OPERATION
PROGRAMMED INPUT OUTPUT (PIO) TRANSMIT AND RECEIVE

ANSWER TO COMPUTER PUZZLE which appeared in THE DEC* PROFESSIONAL Volume 2, Number 6, (November 1983), page 94.




STATEMENT OF OWNERSHIP MANAGEMENT AND CIRCULATION

THE DEC* PROFESSIONAL MAGAZINE

Bi-monthly

Professional Press, Inc. P.O. Box 362 Ambler PA 19477

Carl B. Marbach P.O. Box 362 Ambler PA 19477

James L. Trimbur P.O. Box 362 Ambler PA 19477

Carl B. Marbach 921 Bethlehem Pike Springhouse PA 19477

B.D. Mallory

Helen B. Marbach

Margaret Heine

None

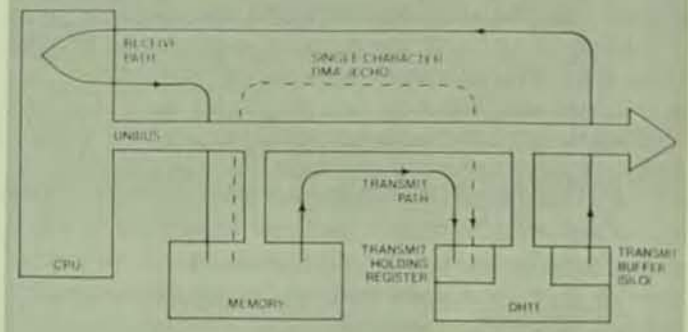
EXTENT AND NATURE OF CIRCULATION	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES (Net Press Run)	80000	85000
B. PAID CIRCULATION	None in U.S.	
1. Sales through dealers and carriers, street vendors and counter sales	XX	
2. Mail Subscription	XX	
C. TOTAL PAID CIRCULATION (Sum of 10B1 and 10B2)	XX	
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	80000	85000
E. TOTAL DISTRIBUTION (Sum of C and D)	80000	85000
F. COPIES NOT DISTRIBUTED 1. Office use, left-over, unaccounted, spoiled after printing	XX	
2. Return from News Agents	XX	
G. TOTAL (Sum of E, F1 and 2 - should equal net press run shown in A)	80000	85000

I certify that the statements made by me above are correct and complete.

SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER

not execute the transfer of data, as is required during Programmed I/O transfers. Although it is a very efficient method of handling large blocks of characters, the initial set-up time required for DMA operations makes it less efficient than Programmed I/O for single character operations, such as character echo. It is for this reason that a DH11 is not the most efficient choice in a data entry application.

FIGURE 2



DH11 OPERATION
DIRECT MEMORY ACCESS (DMA) FOR TRANSMIT AND PROGRAMMED INPUT OUTPUT (PIO) FOR RECEIVE

DMF 32

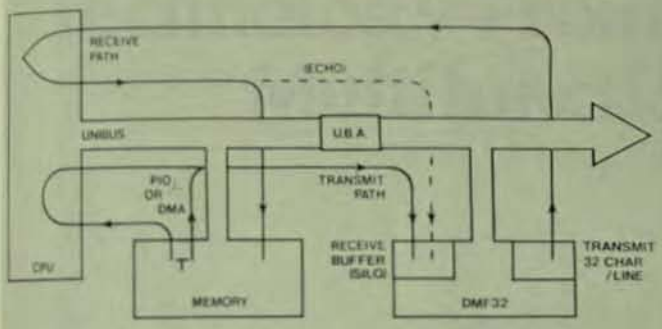
The DMF32 multiplexer combines the best characteristics of both the DZ11 and DH11, while eliminating their respective liabilities. The DMF32 accomplishes this by providing either buffered Programmed I/O or DMA output of characters. Based on the character count of the transfer, efficiency is achieved by selecting either the Programmed I/O method for short character counts, or DMA for longer transfer character counts.

The DMF32 contains a 32 character output buffer for each line to maximize efficiency on Programmed I/O output operations. Using only a single interrupt, the device driver can transfer bursts of up to 32 characters to the output buffer of the controller. The DMF32 then transmits the data from the buffer to the terminal while the CPU is performing other functions. Performance tests on the DMF32 indicate that the CPU time required to transfer two 32 character bursts in Programmed I/O mode is approximately the same as the CPU time required for setting up a DMA operation. For output transfers of 65 characters or more, the device driver selects DMA operations. Therefore, DMF32 efficiency is not application-dependent when outputting, because of the ability to dynamically select either the Programmed I/O or the DMA method of transfer on each individual line.

SUMMARY

As we can see, the DZ11, with PIO output, has the most efficient transfer method for applications handling minimal character count transactions, such as data entry operations. On the other hand, the DH11, with its DMA capability, is not as efficient on single character output, but is more efficient

FIGURE 3



DMF32 OPERATION
PROGRAMMED INPUT/OUTPUT (PIO) OR DIRECT MEMORY ACCESS (DMA) FOR TRANSMIT AND PIO RECEIVE

for block mode output. This suggests that the average PDP-11 user might want to have some combination of both the DZ11 and DH11 for efficient operation.

Since the DMF32 dynamically selects between either PIO or DMA output, depending on the length of data queued, it requires minimal CPU overhead, regardless of the application, while providing the most efficient communications currently available for VAX-11 processor group. The total communication throughput capability of the UBA of a VAX-11/780 with DZ11s is approximately 25,000 characters per second. If the DZ11s are replaced with DMF32s, the total throughput capability is increased to over 100,000 characters per second. So to approach the efficiency of a DMF32, the DZ11 would have to be installed in an extremely application-dependent operation, such as data entry. Once any substantial amount of output is required from a system, the DMF32 quickly becomes two to four times more efficient than the DZ11.

For the PDP-11 Series computers, it might be necessary to segregate applications by using a combination of DZ11 and DH11 where each would benefit overall system operation. With the VAX Series computers, the DMG32 product provides optimized performance regardless of application. The DZ11 may approach the efficiency of the DMF32 in specific applications, such as data entry, and a DH11-type multiplexer, if it were supported on the VAX, might be adequate for tasks requiring significant data output, like word processing. But with its ability to dynamically select between buffered PIO and DMA output, the DMF32 eliminates the need for applications-specific hardware. In achieving this type of versatility, the DMF32 products become the most cost-effective performance enhancement for applications requiring significant terminal communications throughput for VAX computer systems.

CONCLUSION

The question of which communications hardware provides the best performance for your system configuration is a critical one, as we have seen. While cost is an important factor in choosing hardware alternatives, it is maximum throughput that is the critical factor overall.

. . . continued from page 125

FIGURE C

```
      CHARACTER*35 LINE1,LINE2
      CHARACTER*3  ROW1,COL1,ROW2,COL2
C
      CALL CONVERT_NUMBER (12,ROW1)
      CALL CONVERT_NUMBER (12,ROW2)
      CALL CONVERT_NUMBER (40,COL1)
      CALL CONVERT_NUMBER (46,COL2)
C
C             ASSIGN ESCAPE SEQUENCE TO POSITION STRING
C
      LINE1 = CHAR(27)//CHAR(91)//ROW1//CHAR(59)//COL1//CHAR(72)//
     1        'HELLO IBM PROFESSIONAL   '
      LINE2 = CHAR(27)//CHAR(91)//ROW2//CHAR(59)//COL2//CHAR(72)//
     1        'DEC'
C
      WRITE (5,10) LINE1
      WRITE (5,20) LINE2
C
10    FORMAT (' ',A35)
20    FORMAT (' ',A13)
      END
C
C
C
      SUBROUTINE CONVERT_NUMBER (K,BUFFER)
      CHARACTER*3 BUFFER
C
      I = K
      J = I/100                          ! calculate first digit
      BUFFER(1:1) = CHAR(48 + J)         ! convert digit to ascii value
      I = I - (J * 100)                  ! calculate remaining value
      J = I / 10                         ! calculate second digit
      BUFFER(2:2) = CHAR(48 + J)         ! convert digit to ascii value
      I = I - (J * 10)                   ! calculate remaining value
      BUFFER(3:3) = CHAR(48 + I)         ! convert digit to ascii value
C
      RETURN
      END
```

FIGURE D

```
                  column 40
                  |
row 12            HELLO DEC PROFESSIONAL
```

FIGURE E

```
      SUBROUTINE CURSOR( LINE, COLUMN )
C
C-    THIS ROUTINE WILL PLACE THE CURSOR TO A SPECIFIC LINE
C-    AND COLUMN
C
      INTEGER COLUMN
      LOGICAL*1 CHAR(8)
C
C-    PROCESS THE INPUT PARAMETERS
C
      ENCODE ( 8, 1000, CHAR ) '33, '133, LINE, '73, COLUMN, '146
C
C-    SEND THE CONTROL SEQUENCE TO THE VDT
C
      DO 100 I = 1, 8
      IF ( CHAR( I ) .EQ. '40 ) GO TO 100
25    IF ( ITTOUR( CHAR( I ) ) .NE. 0 ) GO TO 25
100   CONTINUE
C
      RETURN
C
1000  FORMAT(2A1,2(I2,A1))
      END
```

FIGURE F

```
      integer function cursor(line,column)

C *
C * VT-100 Cursor addressing
C *
C * Cursor addressing follows the DEC convention of zero-based addressing
C *
C *
C *
C * To keep it somewhat general, the subroutine only checks for
C * negative numbers.
C *
C * The number of characters typed out is returned as a function value;
C * no output is done if an error occurs.
C * Error values are    -1 for negative line number
C *                     -2 for negative column (line must be positive)
C *                     -3 for error converting number (usually the
C *                        number was greater than two digits).
C *

      integer line, column
      integer iline, icol
      character*20    output
      character*1     header(2)
      character*1     trailr
      character*1     seprat
      integer i,j

C * pre-define header, seprat, and trailr strings
      data    header /27, '['/
      data    trailr /'f'/
      data    seprat /';'/

C     Bad line number
      cursor = -1
      iline = line
      if (iline .lt. 0)  return
C     Bad column number
      cursor = -2
      icol = column
      if (icol .lt. 0)   return

C * Build the initial string

C     Bad size or other conversion error
      cursor = -3
      write(output,1234,err=2000)header,iline,seprat,icol,trailr
1234  format(2a,i2,a,i2,a)

C * Make sure there are no spaces in the character string

C     output pointer
      j = 1
C     loop on input -- in-place compression
      do 10 i = 1, 19
        if (output(j:j) .eq. ' ') then
          output(j:) = output(j+1:)
        else
          j = j + 1
        endif
10    continue
      j = j - 1

C * output the result

      type *, output(1:j)
C     status -- positive -- characters output.
      cursor = j

2000  return
      end
```

The Newest Innovation in Controller Technology From the Oldest Name in Multifunction Controllers.

Introducing the World's First LSI-11 Emulating Multifunction Disk/Tape Controller.

Say hello to SPECTRA 25, the cornerstone of our new family of high-performance disk/tape controllers designed for use with DEC's LSI-11 computer.

This revolutionary Q-Bus compatible single quad board lets you interface any combination of two SMD disks and four formatted ½-inch tape drives. By using extended commands to program the onboard E²PROM, you can easily select drive mixing, mapping, and many other features–all without removing the controller from the system.

The SPECTRA 25 emulates DEC's RM02/5 and RM80 disk subsystems, and DEC's TSII tape subsystem. It also provides complete emulation for operation with DEC's RT-11, RSX-11M, RSX-11M-PLUS and RSTS/E operating systems.

To further enhance system performance, Spectra Logic offers SPECTRA STREAM™ software, a streaming tape backup utility that can back up an entire 80MB drive in only seven minutes.

Spectra Logic first introduced the multifunction concept back in 1979. And we've been quietly revolutionizing the market ever since with families of controllers that provide the high-performance, proven reliability, and added value you need to stay competitive.

We also offer the industry's most comprehensive one year warranty, responsive nationwide service, and ongoing technical support.

SPECTRA 25 is the latest innovation in controller technology from the company with peripheral vision. Spectra Logic. For further information, including complete technical specifications, call or write us today.

Spectra Logic Corporation
1227 Innsbruck Drive, Sunnyvale, CA 94089
(408) 744-0930 TWX 910-339-9566
TELEX 172524 SPL SUVL

International Sales Office:
Belgium (32) (2) 5134892

SPECTRA LOGIC

FIGURE G

```
C*****************************************************************
C                                                                 *
C                        * VT52.FOR *                             *
C                                                                 *
C*****************************************************************
C
C       *** VT52 RELATED SUBROUTINES ***
C
C*****************************************************************
C
C               SUBROUTINE VT52 TO CHANGE A VT100 FROM ANSI TO VT52
C               NOTE: Do not use this subroutine on an actual VT52
C
        SUBROUTINE VT52
        IMPLICIT INTEGER(V)
        DIMENSION VT52(5)
        DATA VT52/155,'[','?','2','l'/
C       Change from ANSI to VT52 mode
        WRITE(7,101) VT52
 101    FORMAT(1X,5A1,$)
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE NORMAL FOR RESETING KEYBOARD
C                       (FROM VT52 MODE ONLY)
C
        SUBROUTINE NORMAL
        IESC = 155
C       Get out of graphics and alternate keypad mode
        WRITE(7,101) IESC,IESC
 101    FORMAT(1X,1A1,'G',1A1,'>',$)
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE DCA FOR DIRECT CURSOR ADDRESS FOR VT52 ONLY
C                               I = DOWN        J = ACCROSS
C
        SUBROUTINE DCA(I,J)
        IESC = 155
        ICOL = 31 + I
        JROW = 31 + J
C       Direct cursor
        WRITE(7,101) IESC,ICOL,JROW
 101    FORMAT(1X,1A1,'Y',2A1,$)
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE CLEAR FOR CLEAR SCREEN FOR VT52 ONLY
C               (CLEARS FROM CURRENT LINE TO BOTTOM OF SCREEN)
C
        SUBROUTINE CLEAR
        IESC = 155
        WRITE(7,101) IESC
 101    FORMAT(1X,1A1,'J',$)
        RETURN
        END
```

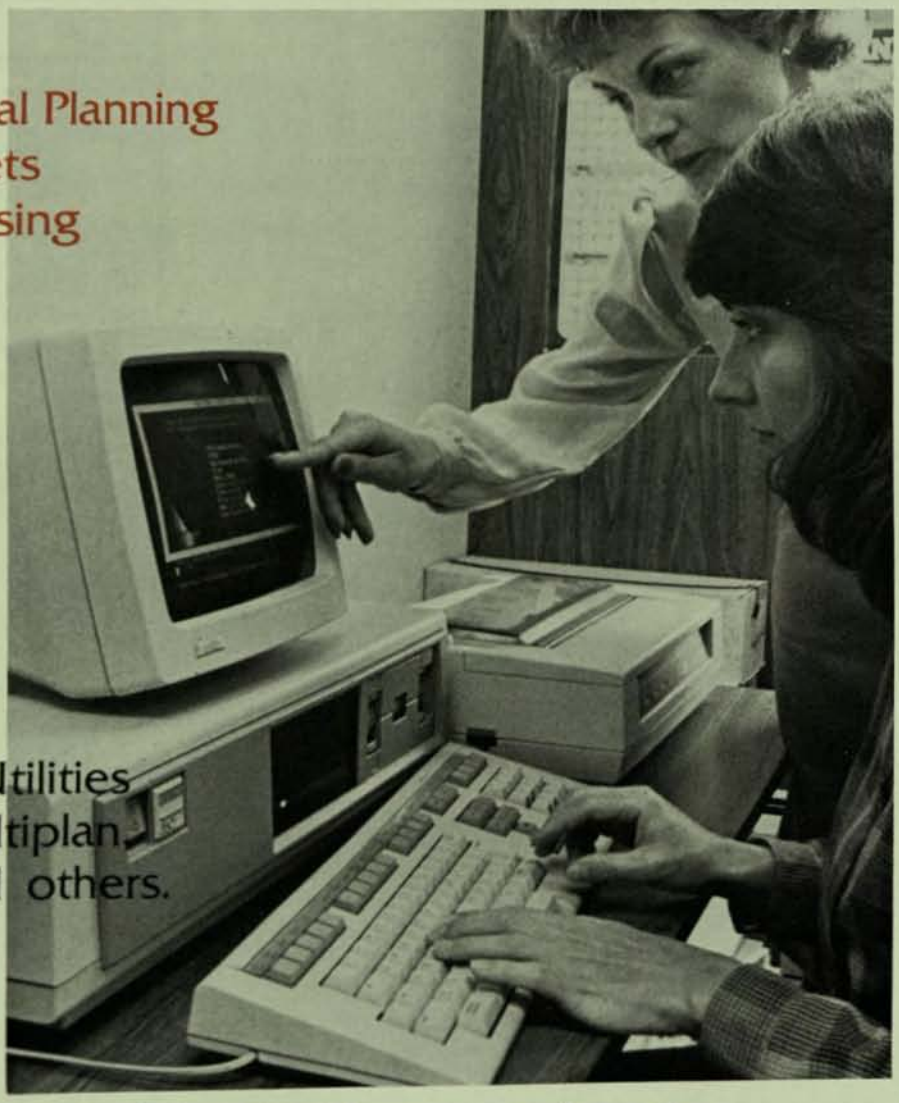
FIGURE H(1)

```
C*****************************************************************
C                                                                 *
C                        * ANSI.FOR *                             *
C                                                                 *
C*****************************************************************
C
C       *** ANSI RELATED SUBROUTINES ***
C
C*****************************************************************
C
C               SUBROUTINE ANSI TO CHANGE FROM VT52 TO ANSI
C
        SUBROUTINE ANSI
        IMPLICIT INTEGER(A)
        IESC = 155
C       Change from VT52 to ANSI mode
        WRITE(7,101) IESC
 101    FORMAT(1X,1A1,'<',$)
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE NORMAL FOR RESETING KEYBOARD
C                       (FROM ANSI MODE ONLY)
C
        SUBROUTINE NORMAL
        IESC = 155
C       Get out of graphics and alternate keypad mode
        WRITE(7,101) IESC,IESC
 101    FORMAT(1X,1A1,'(B',1A1,')B')
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE DCA FOR DIRECT CURSOR ADDRESS FOR ANSI ONLY
C                       (ASSUMES 24 * 80 SCREEN)
C                       I = DOWN        J = ACCROSS
C
        SUBROUTINE DCA(ICOL,JROW)
        IESC = 155
C       Direct cursor
        IF ((ICOL.GT.9).AND.(JROW.GT.9)) GOTO 1
        IF ((ICOL.LT.10).AND.(JROW.GT.9)) GOTO 2
        IF ((ICOL.GT.9).AND.(JROW.LT.10)) GOTO 3
        WRITE(7,104) IESC,ICOL,JROW
 104    FORMAT(1X,1A1,'[',I1,';',I1,'f',$)
        GOTO 10
 1      WRITE(7,101) IESC,ICOL,JROW
 101    FORMAT(1X,1A1,'[',I2,';',I2,'f',$)
        GOTO 10
 2      WRITE(7,102) IESC,ICOL,JROW
 102    FORMAT(1X,1A1,'[',I1,';',I2,'f',$)
        GOTO 10
 3      WRITE(7,103) IESC,ICOL,JROW
 103    FORMAT(1X,1A1,'[',I2,';',I1,'f',$)
C
 10     CONTINUE
        RETURN
        END
```

TRAINING

DECmate II
Rainbow 100
Professional 350

Spreadsheet Financial Planning
Spreadsheets
Word Processing



Microprocessor Utilities
dBase, 1-2-3, Multiplan,
VisiCalc, Select and others.

Classes offered in Philadelphia or your location.
Taught by experienced professionals.

ON TRACK SYSTEMS, INC.
921 Bethlehem Pike, Springhouse, PA 19477 • 215-542-7008

CIRCLE D222 ON READER CARD

FIGURE H(2)

```
C*****************************************************************
C
C               SUBROUTINE CLEAR FOR CLEAR SCREEN FOR ANSI ONLY
C               (CLEARS FROM CURRENT LINE TO BOTTOM OF SCREEN)
C
        SUBROUTINE CLEAR
        IESC = 155
        WRITE(7,101) IESC
 101    FORMAT(1X,1A1,'[J',$)
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE WIDE FOR PRINTING DOUBLE WIDTH CHARACTERS
C
        SUBROUTINE WIDE
        IESC = 155
        WRITE(7,101) IESC
 101    FORMAT(1X,1A1,'#6',$)
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE LARGE FOR PRINTING DOUBLE HEIGHT AND WIDTH
C                       CHARACTERS
C       NOTE: After this call, you still must print the line twice, once
C             for the top half and once for the bottom
C
        SUBROUTINE LARGE
        IESC = 155
C       Set current line to top
        WRITE(7,101) IESC
 101    FORMAT(1X,1A1,'#3')
C       Set next line to bottom
        WRITE(7,102) IESC
 102    FORMAT(1X,1A1,'#4',$)
C       Move back to original line
        WRITE(7,103) IESC
 103    FORMAT(1X,1A1,'8',$)
        RETURN
        END
C*****************************************************************
C
C               SUBROUTINE SINGLE FOR CHANGING A LINE SIZE BACK
C                       TO SINGLE HEIGHT SINGLE WIDTH
C
        SUBROUTINE SINGLE
        IESC = 155
        WRITE(7,101) IESC
 101    FORMAT(1X,1A1,'#5',$)
        RETURN
        END
```

FIGURE I

```
C*****************************************************************
C
C                       S C R E E N . F O R
C
C*****************************************************************
C
C       To demonstrate the screen functions for ANSI
C
C       For a PDP 11/70 (under RSTS/E), the following will compile and
C       run this program (unless you are in DCL mode):
C               FOR ANSI=ANSI
C               FOR SCREEN=SCREEN
C               LINK SCREEN=SCREEN,ANSI
C               RUN SCREEN
C
        CALL ANSI
        CALL NORMAL
        CALL DCA(1,1)
        CALL CLEAR
        DO 1 I=23,1,-1
        IX = I
        DO 1 J=75,1,-5
        JX = J
        CALL DCA(IX,JX)
        WRITE(7,101)
 101    FORMAT(1X,'ANSI')
 1      CONTINUE
        DO 2 I=13,14
        IROW=I
        CALL DCA(IROW,40)
        WRITE(7,102)
 102    FORMAT(41X)
 2      CONTINUE
        CALL DCA(11,1)
        CALL WIDE
        CALL DCA(12,1)
        CALL LARGE
        CALL DCA(14,1)
        CALL WIDE
        CALL DCA(22,1)
        STOP
        END
```

Special Subscription Offer

You can now become a subscriber to PERSONAL and PROFESSIONAL with no obligation or risk.

Simply enter your subscription on this postage-free card and mail it to us. Our current issue will be sent immediately upon receipt.

Read your issue of PERSONAL and PROFESSIONAL, the independent magazine for Digital personal computer users. If it is everything you expected, honor our invoice. If it isn't, just write "cancel" across the invoice and mail it back. You won't be billed and the issue is yours at no charge.

SO ACT NOW. Become a PERSONAL and PROFESSIONAL subscriber today!

PERSONAL and PROFESSIONAL
P.O. BOX 114, SPRINGHOUSE, PA 19477
19542-7008

NAME _____
(please print full name)

ADDRESS _____

CITY _____ STATE _____ ZIP _____

☐ PLEASE BILL ME ONLY $28.00 FOR THE NEXT 12 ISSUES OF PERSONAL AND PROFESSIONAL.

☐ PAYMENT ENCLOSED. I WANT TO GET A BONUS ISSUE! I AM ENCLOSING MY CHECK OR CREDIT INFORMATION.

VISA/MASTER CARD (CIRCLE ONE) EXP. DATE _____

ACCT. # _____

☐ FOREIGN, AIR MAIL, PAYABLE IN U.S. DOLLARS $56.00.

PLEASE SEND BACK ISSUES CIRCLED: V1#1 V1#2 V1#3

PAYMENT ENCLOSED: $3.50 PER ISSUE

BILL ME: $5.00 PER ISSUE

SIGNATURE _____

DP31

hardware and software. . . what industry leaders are saying. . . expert opinion. . . and much, much more!

PERSONAL and PROFESSIONAL is written by DEC users for DEC users. Straight talk, plain talk from the people who know. First hand.

Subscribe today! Special offer of 12 issues for only $20 — a savings of $22 off the cover price. Fill out the postage-paid subscription card and return it immediately. Your copy of PERSONAL and PROFESSIONAL will be sent to you in quick order.

PERSONAL and PROFESSIONAL
P.O. Box 114
Springhouse, PA 19477
(215) 542-7008

CIRCLE 0221 ON READER CARD

VAX/VMS DECNET PROXY ACCESS

By Norm Perron, TRW Defense Systems Group, Redondo Beach, Ca.

DECnet is the name given to both software and hardware that allows Digital computers to be networked together. The implementation of DECnet on VAX computers running the VMS operation system is excellent, as it is built into RMS (VAX Record Management Services- file structure on a VAX) and is available at DCL (DEC Command Language). This implementation does have a problem which causes some difficulty: The problem is that the default access on the remote node is usually not adequate or an Access Control String (Username and password) must be specified. An undocumented, unsupported feature has existed since release 3.0 of VMS to overcome this default access: Proxy Access (formerly called Proxy Login). Before Proxy Access became available, there were two ways to access files on a remote node: Default Access (use default network account) and Access Control String.

An example of Default Access would be if one were to issue the following command at the DCL level:

$ COPY FILE.EXT NODEX::

In this case, the file FILE.EXT would be copied to the remote node named NODEX in its default DECnet account with its UIC and associated privileges, probably not where you would want the file to be copied. If you want the file copied to your account on the second node, you would then log onto that node and copy the file from the default DECnet access account to your own account after possible playing some file protection games, then deleting the temporary file. This requires some extra typing and is frustrating.

The second way to access the network was to use Access Control string such as in the following command:

$ COPY FILE.EXT NODEX"SYSTEM MANAGER"::

Where, SYSTEM is the Username, and MANAGER is the password.

The disadvantages with Access Control String are several: you must expose your password (someone looking over your shoulder can see what has been typed and the information may end up in a command file); it causes extra typing; and the fact that this password is being transmitted over the phone or cable to the second node, reduces overall network security.

Digital realized DECnet access was inefficient and enhanced this software to avoid these two problems. This enhancement is called Proxy Access.

With Proxy Access, if someone were to issue the following command:

$COPY FILE.EXT NODEX::

The file FILE.EXT would be copied directly to NODEX in the directory specified by the network manager, with the UIC and file protection codes established for that user.

Proxy Access Advantages

Proxy Access provides the user with the following advantages:

1. Security. No passwords are needed in Proxy Access because the access is determined at the remote node. In fact, users may access a remote node without even knowing the password for their proxy account on that node. This will avoid transmitting passwords over phone or other lines into which they could be tapped. No one will see passwords being entered on the terminal, and there will not be any passwords in command files.

2. Efficiency. Proxy Access eliminates the need for much typing. There is no need to specify Access Control Strings.

3. Control. The NETUAF.DAT file on the destination node determines which users have access to proxy accounts on this node. It's up to the network manager to determine exactly which users on which nodes are allowed Proxy Access to the local node. Any user may still use the Default Access scheme by merely specifying a null Access Control List as follows:

$ COPY FILE.EXT NODEX"":

Any user may also still use the Default Access scheme by merely specifying an Access Control List as follows:

$ COPY FILE.EXT NODEX"SYSTEM MANAGER"::

How to Set Up Proxy Access

In addition to the changes made to DECnet, Digital made changes to two utilities in their implementation of Proxy Access: AUTHORIZE and NCP.

AUTHORIZE is used to create and/or modify file NETUAF.DAT. This file resides in directory SYS$SYSTEM and contains the remote-node::username to local-user mapping. This indexed file contains several records with the following format:

REMOTE-NODE::REMOTE-USER LOCAL-USER

Where, REMOTE-NODE:: Is the remote node name, REMOTE-USER is the Username to be used on that node and LOCAL-USER is the local Username.

The commands to set up this data base are as follows:

```
$ SET DEF SYS$SYSTEM:
$ RUN AUTHORIZE
UAF>CREATE/PROXY
     ! Create NETUAF.DAT
UAF>ADD/PROXY node::remoteuser localuser
     ! Add an entry
UAF>LIST/PROXY
```

```
        ! which lists all entries
    UAF>SHOW/PROXY node::remoteuser
        ! Show an entry on terminal
    UAF>SHOW/PROXY *
        ! Show all entries on terminal
    UAF>REMOVE/PROXY node::remoteuser
        ! Remove an entry
```

NCP is used to control the overall use of Proxy Access. The following commands will bring up Proxy Access:

```
$ RUN SYS$SYSTEM:NCP
NCP>SET EXECUTOR DEFAULT PROXY BOTH
NCP>PURGE EXECUTOR PRIV USER PASSWORD ACCOUNT
```

In this example, BOTH means incoming and outgoing access is allowed. The BOTH parameter may be replaced with INCOMING, OUTGOING or NONE depending on what type of access is desired.

An Example of Setting Up Proxy Access

1. Create a CONFIGNET.COM file for NODEA similar to the following:

```
$!CONFIGNET.COM
$ run sys$system:sysgen
connect net/noadapter/driver = netdriver
$ run sys$system:ncp
purge known lines all
purge known circuits all
purge known logging all
purge known nodes all
purge known objects all
define executor        address 2-
                       buffer size 576-
                       maximum address 5-
                       maximum buffers 45-
                       maximum circuits 32-
                       state on
define executor        name NODEA-
                       nonprivileged-
                           user netnonpriv-
                           password nonpriv
                       !Note: No need for Priv Account
define node 1          name NODEB
define line dmc-0      protocol ddcmp point-
                       receive buffers 8-
                       state on
define circuit dmc-0   cost 4
                       service enabled-
                       state on
define executor default proxy both!Turn Proxy On exit
```

2. Modify the previous CONFIGNET.COM file for NODEB as follows:

Change address to 3

Change define executor from NODEA to NODEB

Change define node 1 name NODEB to define node 2 name NODEA

3. Execute your CONFIGNET.COM file on both nodes as follows:

```
$ @CONFIGNET
```

4. Shut down DECnet on both nodes if not already shut down as follows:

```
$ MCR NCP SET EXECUTOR STATE OFF
```

5. Start up the new network on both nodes as follows:

```
$ @SYS$MANAGER:STARTNET
```

6. Create a few Proxy Access accounts on both nodes as follows:

```
$ SET DEF SYS$SYSTEM
$ RUN AUTHORIZE
UAF>CREATE/PROXY
UAF>ADD/PROXY NODEA::USER1 USER1
UAF>ADD/PROXY NODEA::USER2 USER2
UAF>ADD/PROXY NODEB::USER1 USER1
UAF>ADD/PROXY NODEB::USER2 USER2
UAF>EXIT
```

Note, there are entries entered for USER1 on both the local (NODEA) and the remote node (NODEB). There are two reasons for this. First, users sometimes use network utilities located on the local node. Second, a network may consist of the same users having an account on every node in the network. This way, only a single copy of NETUAF.DAT needs to be maintained and then it could be copied to each node in the network.

7. Log onto NODEA as USER1 (or any Username you may have substituted in Step 6 above).

8. Try accessing your files on the remote node as follows:

```
$ DIRECTORY NODEB::
```

You should get a directory of [USER1], not the default account. If you get the default account, something went wrong. One fix I have found, is to rename all .DAT files in SYS$SYSTEM starting with NET to a TMP extension as you may have had some strange things set up in the old network data files. An example of this is as follows:

```
$ SET DEFAULT SYS$SYSTEM:
$ RENAME/LOG NET*.DAT;* *.TMP;*
```

After you've completed bringing up Proxy Access, these files should be deleted.

Then start again with Step 3.

9. If it still doesn't work, try the following:

```
$ MCR NCP PURGE EXEC PRIV USER PASSWORD ACCOUNT
$ MCR NCP DEFINE EXEC PROXY BOTH
$ MCR NCP PURGE NODE XXX PRIV USER PASSWORD ACCOUNT
$ MCR NCP PURGE NODE XXX NONPRIV USER PASSWORD ACCOUNT
$ MCR NCP PURGE NODE XXX PROXY BOTH
```

Where, XXX is for each remote node.

```
$ MCR DEFINE KNOWN OBJECTS PROXY BOTH
$ MCR NCP DEFINE OBJECT TASK OUTGOING
```

I did not have to do any of these NCP commands, but if your network is complicated, you may have to issue some of these commands.

Summary

The reason Proxy Access is not documented or supported at this time is because it is still experimental. Digital is now using this feature in-house to see what types of problems it may create. One potential problem is that the network becomes as secure as its least secure node. There have been sessions at DECUS since Fall '82 about Proxy Access, and it appears most users are quite pleased with its features. At the Fall '83 DECUS, a session on "Mechanics of Proxy Access" brought up the question "Will Proxy Access be supported in the next major release of VMS?" The Digital speaker commented that no decision had yet been made. Since it is used quite heavily in the 1000+ nodes on Digital's engineering net, I feel it will be supported in the near future. If not, it should definitely be available in unsupported fashion, and operate basically as I have described.

FITTING FILES ON FLOPPIES
(RT11 Directories on Floppy Disks)

By Austin Kinsella, Regional Technical College, Kilkenny Road, Carlow, Ireland

1.0 INTRODUCTION

If your RSTS/E system is fitted with RX single or double density disk drives, there are several ways of storing files on the floppies. One way, of course, is to use BASIC+, but since RSTS does not recognize a directory structure on the floppies, you must open the device non-file structured, and do your own directory work. Another is to use one of the CUSPs supplied by DEC, such as FIT or FLINT. In the Regional Technical College in Carlow, students use FIT to store low-activity files on double density floppies. To reclaim scattered free space, it is necessary to do regular copies of full floppies, a process at which FIT is rather slow, so it was decided to write a fast floppy copy utility. Written in MACRO, with a 100 block buffer, the program copies a 50 file, 900 block floppy in about 90 seconds. The directory structure which FIT uses on floppies, as learned from the RSTS Users Guide, FIT listings, and liberal applications of DSKDMP, is described in this article.

2.0 GENERAL STRUCTURE

FIT uses the RT11 directory system, so disks written by FIT are directly usable on RT11 systems, and FIT can read disks from an RT11 system. In an RT11 directory, all space on the disk is recorded in the directory, either as blocks allocated to files, or blocks available for use. Because files are contiguous, for any file or free space all that is needed for access is the starting block number and number of blocks. The entries in the directory are in sequence of starting block number. The files (or free space) start immediately after the disk header and the directory space. Files are written in the first sufficiently large free space, leading to fragmentation of free space as deletions take place.

3.0 HEADER

The header occupies blocks one through five, but only the first block has anything in it. The first two bytes of this block must be zero.

At byte offset 472 (i.e., starting with byte 473) is a volume identification, 24 bytes long. At byte offset 496 (i.e., starting with byte 497) is a 16 byte format type. FIT uses 'RSTS70' as the volume identification, and 'DECRT11A' as the format type, but does not check these fields on input floppies.

4.0 DIRECTORY

The directory consists of a pre-allocated number of two-block clusters, starting at blocks six and seven. By default,

FIT allocates 10 clusters when it zeroes a floppy, though the user may specify from one to 31, and the default is dependent on the file entry size and the device size. Each cluster in use contains a cluster header, and at least one directory entry. On an empty floppy only the first cluster is used, with a single entry showing that all blocks are free.

4.1 CLUSTER HEADER

The cluster header contains five words; as detailed in Figure 1.

4.1.1 Next Cluster

The clusters in use are not necessarily sequential. If a new file is to take up part of an unused space on the floppy (caused by an earlier deletion) two directory entries will be needed, one for the new file, and one for the remaining unused blocks, where previously there was only the one for the unused blocks. If the cluster was full of entries, it must now be split over two clusters to accommodate the extra entry. If the split cluster was not the last cluster in use, the clusters are no longer sequential. It is thus necessary in each cluster to maintain a pointer to the next cluster.

4.1.2 Extra Words

The RT11 directory system can handle a range of directory entry sizes. The minimum entry is seven words. Additional words can be allocated at initialization (zeroing); by default FIT allocates 13 extra words for a total entry size of 20 words, and uses these extra words to record runtime system and file attribute information. If the number of extra words is other than 13, or the words contain information not considered by FIT to be reasonable, they are ignored.

FIGURE 1

Cluster Header	
Word	Usage
1	clusters allocated for the directory
2	cluster number of next cluster
3	clusters in use (only maintained in 1st cluster)
4	extra bytes of information in directory entry
5	block number of 1st file this cluster

FIGURE 2

7 Standard Entry Words	
Word	Usage
1	low byte reserved, high byte entry status
2-3	file name, radix 50
4	file extension, radix 50
5	filesize
6	job, channel number (not used by FIT)
7	creation date (RT11 Format)

4.1.3 Start Block Number

Because files and free space are stored in block number order, and all blocks are recorded, it is not necessary to store a file start position for each file entry, so only the start position of the first file in each cluster is recorded. Programs accessing the directory must add on the file sizes as they step through the cluster to get the start block number of the entry of interest.

4.2 DIRECTORY ENTRY

The word allocation of a directory entry is set out in Figures 2 and 3. The number of entries that a cluster can accommodate depends on the extra words allocated. With the default of 13, a cluster will take 25 entries, leaving 14 unused bytes at the end of the cluster. An entry may span the block boundary within the cluster. With the default of 10 clusters, FIT will have

FIGURE 3

```
        13 Extra Entry Words

Word        Usage
 8          record format, file organization,
            carriage control and block spanning
 9          record size
10-11       file size
12-13       blocks in use
 14         byte count, last block
 15         bucket size and header size
 16         maximum record size
 17         extension size
 18         not used
19-20       RTS name (radix-50)
```

FIGURE 4

```
            Entry Status
Byte Value  Word Value    Meaning
  1             256       Tentative file
  2             512       Unused blocks
  4            1024       Normal
  8            2048       End of data
```

```
1  EXTEND
1000      OPEN 'DYO:' AS FILE 1%, RECORDSIZE 1024%                          &
\  PRINT TAB(7);'Basic Files on DYO:'                                        &
\  PRINT 'Name          Start         Size'                                  &
\  NEX.CLU% = 1%                                                            &

1010      WHILE NEX.CLU%                                                    &
\         END.CLU.FLAG% = 0%                                                &
\         GET  1%, RECORD 4% * NEX.CLU%*2%                                  &
\         FIELD  1%, 2% AS CLU.MAX%,                                        &
                     2% AS NEX.CLU%,                                        &
                     2% AS CLU.IN.USE%,                                     &
                     2% AS ENT.EXT%,                                        &
                     2% AS START.BLOCK%                                     &
\         CLU.IN.USE% = FNCI%(CLU.IN.USE%)                                  &
\         NEX.CLU% = FNCI%(NEX.CLU%)                                        &
\         START.BLOCK% = FNCI%(START.BLOCK%)                                &
\         ENT.SIZ% = 14% * FNCI%(ENT.EXT%)                                  &

1020            FOR ENT.CNT% = 1% UNTIL END.CLU.FLAG%                       &
                        OR (10% * ENT.CNT%*ENT.SIZ% > 1024%)                &
\               FIELD 1%, 10% AS HEAD%,                                     &
                          ENT.SIZ%*(ENT.CNT%-1%) AS DUMMY%,                 &
                          1% AS UNUSED%,                                    &
                          1% AS FIL.STAT%,                                  &
                          2% AS NAM1%,                                      &
                          2% AS NAM2%,                                      &
                          2% AS EXT%,                                       &
                          2% AS FIL.SIZ%                                    &
\               FIL.SIZ% = FNCI%(FIL.SIZ%)                                  &
\               FIL.STAT% = ASCII(FIL.STAT%)                                &
\               PRINT FNCR$(NAM1%);                                         &
                      FNCR$(NAM2%),                                         &
                      START.BLOCK%,                                         &
                      FIL.SIZ%                                              &
                        IF FIL.STAT = 4% AND FNCR$(EXT%) = 'BAS'            &
\               START.BLOCK% = START.BLOCK% + FIL.SIZ%                      &
\               END.CLU.FLAG% = FIL.STAT% AND 8%                            &
\         NEXT ENT.CNT%                                                     &
\  NEXT                                                                     &
\  GOTO 32767                                                               &

15000     DEF* FNCI%(ARG%) = SWAP%(CVT%%(ARG%))
15010     DEF* FNCR$(ARG%) = RAD$(FNCI%(ARG%))                              &

32767     END
```

space for up to 250 files. If a few large files are required, more file space can be made available by allocating only a single directory cluster, by using /N:1 when zeroing.

4.2.1 Entry Status

The high byte of the first word of each entry gives the status (the low byte is currently unused). Tentative files will not normally be found, unless an error has occurred after the directory entry has been made and before the completion of data transfer. Unused blocks are recorded as type two, and only the file size in the entry has significance. Tentative files should normally be treated as unused. A dummy entry in each cluster follows the last real entry to flag the end of entries in the cluster if there are fewer entries than will fit in the cluster.

5.0 EXAMPLE

As an example of accessing FIT-created directories, at the end of the article is a short (no comments!) BASIC+ program which will list the name, position and size of all files with extension BAS on a floppy. ■

WHAT'S NEW FOR '84

DECspell

DEC announced a speller for the DECmate. It is Z80 based, rather than PDP8 code and operates in a really neat co-processor mode with WPS in the DECmate. DECspell uses a 71,000 word American Heritage (©) dictionary and a personal dictionary for your favorite technical terms. DECspell is phonetic. That means you can make outragous mispellings and the program will still make meaningful suggestions for the correct word. The photo is worth a thousand words (71,000 words?).

The DECmate is experiencing some amazing changes. The new operating environment switches between operating systems as fast as we are accustomed to switch between programs. Some significant enhancements have been made to good old WPS, including full support for the arrows, find, select, insert, previous screen, etc. keys on the LK201 keyboard. Other enhancements include required space, status line and go-to and delete ruler commands. The DECmate now supports a 10MB Winchester and includes read/write passwords to increase document security. This new release of WPS is known as Version 1.5

The DECmate has also acquired a graphics option. When in the VAX ALL-IN-1 environment, you can produce graphics via DECslide and DECgraph. The four plane graphics board can drive either the black and white monitor or the color VR241.

DECMATE FEATURES SIXTEEN COLOR GRAPHICS

Digital Equipment Corporation's VAX/ALL-IN-1 graphics applications, such as DECslide and DECgraph, now may be developed on the DECmate Office Workstation. By emulating a VT125 monitor, the Office Workstation displays as many as 16 colors. Output can be directed either to a VR201, the standard black and white monitor (right), a VR241, and optional color monitor (left), or both.

DECtalk

'An audio printer and touchtone to ASCII converter.'

DEC has announced a simple but very sophisticated product that should expand the way we can deal with our machines. DECtalk is a text-to-speech system that is self contained, can live between your terminal and the computer port, can handle an attached phone line, and basically translates simple ASCII text into understandable speech.

The box can generate a wide variety of voices (male / female / child) and vary its rate of speech from 120 to 350 words per minute. It has an unlimited vocabulary, since it generates phonemes from the ASCII characters. You can define and down-load a user dictionary for trade terms, acronyms and other special words. All speech parameters are tunable to produce custom voices.

The box can answer the phone, offer a verbal menu of choices to the caller, accept and translate the caller's touch-tone response and pass it to the host. DECtalk is controlled by escape sequences from the host. These sequences control every aspect of operation, speech control and phone management.

At the press conference announcing DECtalk, we were given a demo of MCI's mail system. You can call an MCI node (a VAX, of course) from anywhere in the country and have your mail "read" to you by a DECtalk box. You can store pre-written responses and have them sent by a touch-tone option.

DECtalk opens many new vistas for the visually handicapped. Since DECtalk can "read" ASCII text, and since almost all publishing is now done in ASCII text, it should be possible to make magazines available in ASCII on floppies. Talking terminals have been built and are in use. (There is a DEC-internal talking terminal program for the Rainbow in use at the Atlanta phone support center.) A program is necessary to handle the many embedded escape sequences for cursor control and highlighting that litter your screen-filling programs. The program can "echo" the letters you type, or "echo" words. (A word is defined as delimited by spaces, tabs or carriage returns . . .)

DECtalk costs $4000 and should be available as you read this.

ALL-IN-1 Upgrades

DECpage

DECpage is a system to produce high quality office documents including letters, reports, memos and overheads from an LN01 laser printer. No text markup, graphic art skill or knowledge of page layout, fonts or codes is needed.

Seven format types and 48 print styles are available. DECpage also provides vertical and horizontal justification, pagination, some custom print styles and graphic logos.

DECdx

DECdx is a utility to transfer documents between the VAX and the DECmate workstation. DECdx is a menu driven, document exchange facility that allows free movement of documents while preserving the content and format.

The ALL-IN-1 Handbook

DEC has published a handbook covering the full spectrum of their Office Information Systems product. This volume covers all aspects of Digital's office offerings:

The User Interface

Office Capabilities
- Word Processing
- Graphics
- Voice and Electronic Mail
- Calculation and Modeling
- Time and Resource Management
- Filing and Retrieval
- Movement of Information

Data Processing Capabilities
- Application Development
- Information Management
- Office Networks

Systems, Workstations, Storage and Printers
- VAX & PDP-11 System Families
- Workstations
- Storage Devices
- Printers

All in all, an excellent reader for those wishing to view the breadth of DEC's Office offerings.

To receive a copy, just circle D231 on the reader service card.

DECPROFESSIONALDECPROFESSIONALDECPROFESSIONALDECPROFESSIONALDECPROFESSIONALDE ONALDECPROFESSIONALDECPROFESSIONALDECPROFESSIONALDECPROFESSIONALDECPROFESSIONA

PROFESSIONAL

^ etc

educational
timely
Communications

NOTES
MEETINGS
NEWS RELEASES
NEW PRODUCT INFORMATION

Send Notes, Meetings, News Releases, New Product Information, etc. to: PROFESSIONAL etc, P.O. Box 362, Ambler, PA 19002-0362.

CONTENTS

Dear DEC User:

Professional et cetera is your section for educational, timely communications. It is a place for you to announce upcoming events or meetings and a place to comment on them.

It is a place for you to read about new products, and a place to comment on them.

It is a place for you to hear from companies whom you service or who service you — and a place to comment on them.

Please send all material to: Professional etc, The DEC* Professional, P.O. Box 362, Ambler, PA 19002-0362. All announcements must be in our office one month prior to publication.

The Publishers

PROFESSIONAL etc

NEW PRODUCTS

ILS SOFTWARE NOW IN SUPERCHARGED WORKSTATION PACKAGE

Signal Technology, Inc.'s ILS (Interactive Laboratory System) is now available as a workstation package in binary form for Digital Equipment Corp.'s MICRO/PDP-11 Computer. Cost for the package, according to STI Vice-President, Dr. Larry Pfeifer, is 40% less than standard ILS in source form.

The new package is referred to as "Workstation ILS" by the designers and is offered for either RSX-11M/M-Plus or RT-11 operating systems.

According to Dr. Pfeifer, ILS and the MICRO/PDP-11 team up to make an engineering workstation which is a breakthrough in convenience, versatility and quality. The compact DEC system fits on or under a desk, yet offers 16-bit architecture, 10Mb of Winchester disk storage, and the power to support up to 10 terminals.

A new "Supercharged" signal processing capability is also available from Signal Technology, Inc. (STI). It consists of integrating STI's ILS signal processing software package with Sky Computers, Inc.'s array processor and DEC's MICRO PDP-11/23 micro computer.

ILS-SKY allows use of the array processor without programming. Also, the package offers interactive graphical digital signal processing usually offered only on large or dedicated systems. All ILS programs requiring intensive arithmetic calculations utilize the SKYMNK whenever possible. ILS-SKY uses FORTRAN virtual arrays to minimize disk read/write overhead, and is distributed in source or binary form.

Prices range from $4,500 to $15,500 depending on operating system and distribution form (source or binary). Details are available by contacting Dr. Pfeifer at Signal Technology, Inc., 5951 Encina Road, Goleta, CA 93117. Telephone (805) 683-3771. Outside California, the toll-free number is (805) 235-7587.

WORKSTATION SIGNAL PROCESSING. ILS, the world-standard signal processing software package from Signal Technology, Inc., is now provided in binary form for DEC's MICRO/PDP-11. The system allows users to process and analyze signals without programming.

TIME-SAVING SCRIBE 11 INTRODUCED BY WESTEC

Westec Associates, Inc., of Bellevue, Washington, announces the release of the new program application system, SCRIBE 11.

SCRIBE 11 is a software tool designed for application programmers as a reliable and efficient method of utilizing the full set of graphic capabilities offered by the VT 100 family of terminals, including AVO, (Advanced Video Option). SCRIBE 11 allows the generation of user-friendly application screens.

SCRIBE 11 runs on DEC PDP-11 computers under RSX 11M and RSTS/E callable from BASIC PLUS 2 and FORTRAN routines. Specialized coding is eliminated via single instruction cells. The over forty macro routines include: cursor control, line and screen erasing, tab setting, keypad selection, histograms, bar charts, special graphic characters, screen modes and scrolling.

The SCRIBE 11 can be used to write all software programs such as forms management, word processing, spreadsheets, and graphs. The SCRIBE 11 program sells at $995. For more information, contact Westec Associates, Inc., 1607 116th Avenue, N.E., Suite 108, Bellevue, Washington 98004, (206) 454-2901.

IBM PERSONAL COMPUTER LINKED TO UNIX ON DEC MINICOMPUTERS

Interactive Systems Corp. has announced the INtext II Emulator, a new software package that allows the IBM Personal Computer to act as an intelligent terminal for the company's enhanced versions of the UNIX operating system.

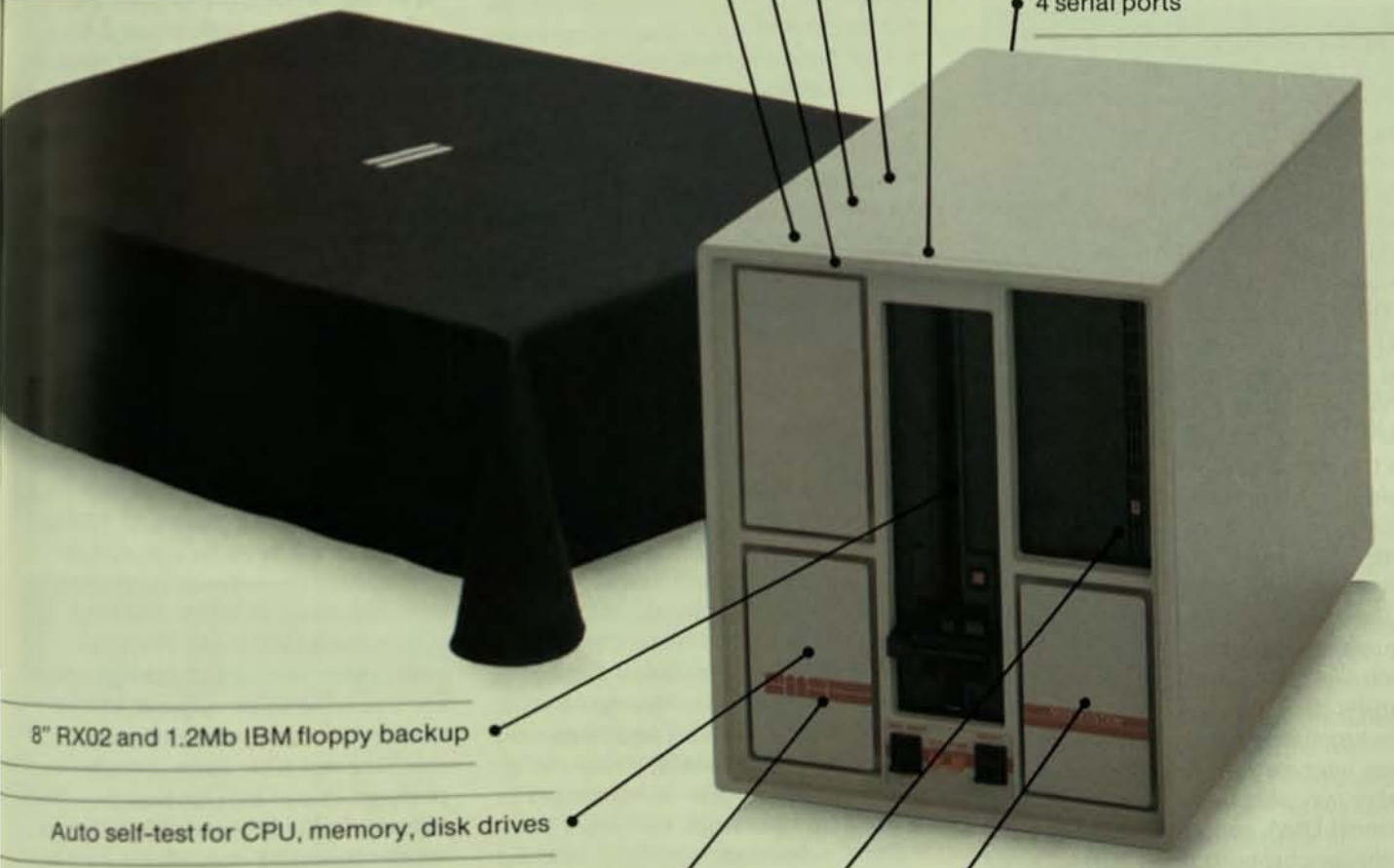
Interactive has designed the INtext II Emulator for PC users who want a single device for performing normal IBM PC functions, as well as communicating with Interactive's UNIX-based editor, INed; Interactive's UNIX-based Advanced Productivity Systems; DEC VMS software; and most public computer networks.

With the emulator, the IBM PC can also function as a VT100 or VT52, so that it can access the large body of sophisticated screen software that has been developed for these devices. It can also function as a standard "dumb" terminal to access asynchronous communications ports on most machines and most public communications networks.

40Mb Winchester Disk

Available now!
And SMS is not afraid to show it

SMS' MDX-11
Micro Winchester
System in a Micro Package



- Compact 9.5"W x 11.2"H x 17"D table top package
- 256Kb main memory (4Mb addressability)
- Expansion space for 4 dual LSI-11 boards
- PDP-11* architecture compatible with DEC* RT-11, RSX-11M, Unix** and TSX+*** software
- LSI-11/23 processor
- 4 serial ports
- 8" RX02 and 1.2Mb IBM floppy backup
- Auto self-test for CPU, memory, disk drives
- Easy access to LED display of system faults
- 10.6Mb, 15.9Mb or 35.6Mb 5¼" Winchester disk
- Advanced disk controller with ECC, flaw management, bootstrap, error retry, and direct non-interleave data transfer
- AVAILABLE WITHOUT CPU and memory for volume requirements

SMS

Scientific Micro Systems, Inc.

777 E. Middlefield Road
Mountain View, California 94043
(415) 964-5700

AUTHORIZED SMS DISTRIBUTOR FOR DEC Q-BUS PRODUCTS: FIRST COMPUTER CORPORATION (312) 920-1050

SMS SALES OFFICES:
Seattle, Washington (206) 883-8303
Boston, MA (617) 246-2540; **Atlanta, Georgia** (404) 296-2029; **Morton Grove, Illinois** (312) 966-2711; **Yorba Linda, California** (714) 993-3768.

*Trademark of Digital Equipment Corporation **Trademark of Western Electric.
***Trademark of S & H Computer

CIRCLE D217 ON READER CARD

PROFESSIONAL etc

NEW PRODUCTS

The INtext II Emulator is available immediately at a unit price of $475, with discounts for quantities exceeding ten units. Interactive Systems Corporation is based in Santa Monica, Calif., with regional offices in New York, Dallas, San Francisco and Washington, D.C. For further information, telephone: (213) 450-8363.

SYNCHRONOUS SERIAL BOARD WITH DMA OPTION

The Model 315 from Grant Technology Systems Corporation, Chelmsford, Massachusetts, is a single channel serial I/O port for interfacing DEC's Q-Bus to synchronous modem compatibility.

The 315 directly replaces DEC's DUV-11. It is compatible with both the software and diagnostic software which is used by DEC to support the DUV-11. The Model 315 is packaged on the standard 8.9" x 5.2" DEC style dual height board and is electrically and mechanically compatible with DEC's LSI-11, 11/2, and 11/23 Microcomputer Series.

The Model 315 may optionally be equipped with a two channel DMA controller which may be used to transfer data directly to and from the synchronous serial port without processor intervention. The DMA controller may also be used as a two channel DMA controller independent of the synchronous serial port.

The Model 315 consists of a synchronous transmitter and receiver in which the various modes of operation are program controlled. These modes include synchronous or isochronous character transmission and reception, character length, parity, sync character, and full or half duplex.

The Model 315 lists for $695 in quantities of 1-9; delivery is stock to 30 days. For further information contact Mr. Steven Bakalis, at Grant Technology, (617) 256-8881.

FIXED ASSET DEPRECIATION SYSTEM

Resource Systems Corporation of Winston-Salem, North Carolina, announces its fixed asset system for the Digital VAX/VMS operating system.

Highlights include:

- Conventional depreciation methods
- Straight line
- Declining balance
- Double declining balance
- Sum of Year Digits
- ACRS
- Automatic conversion from accelerated depreciation methods to straight line depreciation
- Full or partial disposals
- Accommodates asset transfers between departments
- Includes 1983 tax law changes.

Reporting includes:

- Report Writer
- Files are compatible with Datatrieve, a report writer available for Digital (DEC). Domain definition included
- 11 reports — Fixed Asset Analysis, Fixed Asset by Location, Asset Traffic Report, Insurance Value Report, Annual Depreciation Report, Investment Tax Report, Yearly Disposal Asset Summary and Summary for Selling Purposes.

Operations include:

- Multiple operators
- Entry validation
- Two level password security
- Software table driven.

For more information contact Resource Systems Corporation, P.O. Box 1546, Clemmons, N.C. 27012, (919) 766-0571.

TURN YOUR LA50 INTO AN X,Y PLOTTER

A package, by the Software Group of VR Systems, allows users of the versatile DEC LA50 printer to take full advantage of its plotting capability.

The RSX11M based plotting package comes with FORTRAN-77 and BASIC+2 callable subroutines and a plotting utility. All three are capable of exploiting the 1152 dot horizontal resolution of the LA50.

An 8 by 11 inch page image file holds all of the user's plotting requests. Pieces of the image file can be selectively copied, erased, or "drawn on" using the plotting utility or either of the subroutines provided.

Full vectored point-to-point plotting, with simulated "pen" control, is supported with user selectable origin and data scaling. The user simply specifies an origin and scale that allows the data to fit on an 8 by 11 inch page of the LA50. The image file can be "drawn on" by several different users or programs, each using a different section or overlapping sections. Once all the desired images have been drawn, the plotting utility will create a compressed ASCII file of escape sequences capable of driving the LA50. This output file can be included in user text files using standard DEC editors such as EDT. This feature also permits more than one output file to be printed consecutively making for a virtually endless plotting surface.

Send your RL02 pack and $79, or $269 and VR Systems will include the RL02. The software is primarily distributed via RL02 for RSX11M systems with RMS and FPU; however, other media and configurations are available upon request. While detailed documentation comes in machine readable form with the package, it can be purchased separately for $10 per copy. Arizona residents add 6% tax. Please specify F77 or BP2 when ordering. VR Systems 7404 N. 51st Ave., Glendale, AZ 85301, (602) 242-9401.

MEMORY EXPANDER EXTENDS REAL-TIME CAPABILITIES OF DEC FALCON SBC-11/21

Infosphere, Inc. today announced its FMX-11 CMOS (Complementary Metal Oxide Semiconductor) memory expander for the DEC Falcon SBC-11/21.

The FMX-11 extends the Falcon's

on-board memory capability to 60 kilobytes, adds battery support for CMOS static random access memory (RAMs) and provides a CMOS calendar clock and power failure-detection logic with battery backup. These additions make the Falcon well-suited for dedicated, embedded industrial-control applications.

OEM and end-user system integrators can use the FMX-11 to develop and execute real-time applications in process control, instrumentation, data acquisition and energy management. The FMX-11 becomes a support component in the implemented system.

The FMX-11 is being shipped in quantity this month and is priced at $700.

Infosphere, Inc., based in Portland, focuses on hardware and software development methods and tools to support real-time applications. For more information, contact Infosphere, Inc., 4730 S.W. Macadam Ave., Portland, OR 97201, (503) 226-3515.

VIRTUAL DISK SOFTWARE PERMITS STAND-ALONES TO SHARE DISK RESOURCE

Computrol Division of Kidde Automated Systems, Inc. has introduced "virtual disk" software that permits RT-11 and RSX-11M/S based real-time application programs running in stand-alone processors to share the disk resources of a host computer that is running other applications. A full disk-based operating system like RSX-11M can run at the remote site without disks. The power-up bootstrap is performed automatically across the network, eliminating the need for ROM-based satellite software. The satellite stations are linked with the host computer via a 1 Mbps local network implemented with Computrol Megalink processor to processor communication hardware. A measured data throughput of 600K bits per second has been demonstrated in a network consisting of a PDP-11/23 host, and three LSI-11/02 satellites.

Individual floppy disks at the satellite processors are replaced by files on the host computer disk, permitting the satellites to be sited in environments too severe for reliable disk operation. Network operations are transparent to the satellite application programs, which continue to make standard I/O file requests. Thirty-two satellites can be accommodated, and host applications run concurrently with only slight loading caused by network activity.

Prices start at $1,000.00 — immediate availability. For further information about virtual disk software and Megalink 1 Mbps interprocessor links contact John Ricketson, Computrol/Division of Kidde Automated Systems, Inc., 15 Ethan Allen Highway, Ridgefield, CT 06877-6297, (203) 544-9371.

DISTRIBUTED UNIX SUPERMICRO WITH MAINFRAME CAPABILITY

CADMUS Computer Systems, a high-technology startup, recently announced the CADMUS 9000, a family of supermicrocomputers capable of expansion to accommodate over 1000 users in a shared-resource distributed UNIX environment. Aimed at replacing minicomputers and mainframe installations, the CADMUS 9000 systems also offer high-resolution graphics terminals and a variety of other professional workstation features.

CADMUS 9000 systems are based on Motorola Inc.'s MC68000 microprocessor and Digital Equipment Corp.'s Q-Bus. UNISON, CADMUS' proprietary networking software, combines the resources of the individual nodes into a distributed UNIX environment, either over the industry standard Ethernet or a high-performance 50 Megabit/second fiber optic net.

CADMUS Computer Systems designs, manufactures and supports advanced function workstations and supermicrocomputer systems for the high-performance, distributed UNIX environments. CADMUS is based in an historic manufacturing facility in Lowell, Massachusetts, and is to occupy at first 30,000 sq. feet of the structure, currently being renovated for high technology companies. Further information is available from Cadmus Computer Systems, 600 Suffolk Street, Lowell, MA 01854.

PROOF V2.2 SPELLING CHECKER AND CORRECTOR!

Profiteer Company has announced the release of an optimized program to assist in the proofreading process. PROOF is an automated tool which has been refined into a general purpose program. It is compatible with all word processors and text file formats. The program runs under DEC's RT-11 and S&H's TSX (other OSs to follow) with a minimum of 19K words of user task memory space and has been optimized for speed. The PROOF program will run on a system configured with a minimum of two single density floppy drives.

The proofreading program distribution contains a compressed 50,000 word dictionary.

The purchase of a license includes the cost of all future versions with the exception of nominal media and handling charges. PROOF is available for a single user license price of $150.00. For further information, contact Profiteer Co., 7855 Colonial Village Row, Annandale, VA 22003, (703) 560-4784.

FOUR-COLOR PRODUCT LITERATURE DESCRIBES GRAPHIC DISPLAY TERMINAL

New four-color product literature that describes the high-performance graphics and interactive display capabilities of the *concept* GVT Graphic Display Terminal is

NEW PRODUCTS

available, without charge, from Human Designed Systems, Inc.

The terminal, which offers a practical resolution for business and analysis graphics requirements, is illustrated and summarized in the two-page literature. Its capabilities in both graphics and interactive display modes are detailed.

Features of the terminal, including its ANSI X3.64 functionality, Tektronix 4010/4013 compatibility, DEC VT100 software compatibility, and switchable 80/132 column capability are also described. The literature is available from Human Designed Systems, Inc., 3440 Market Street, Philadelphia, PA 19104, (215) 382-5000.

CORTEX FACTORY GENERATES APPLICATION POST-HASTE FOR BUTLER MANUFACTURING

As part of the initial CORTEX Application Factory training course, Butler Manufacturing system analysts developed in only 1½ days a customer tracking application for nationwide distribution.

The CORTEX Application Factory automates developing online business applications for Digital Equipment Corporation minicomputers, microcomputers, and superminicomputers. System designers compress months of conventional programming development into days using the FACTORY because designers need only specify — not program — applications in order to build them. With the FACTORY, developers specify systems simply and entirely through video dialog. From these specifications, FACTORY constructs operational applications complete with screens for data entry/update/retrieval, reports, computation, and data base.

The CORTEX Application Factory is now available for $15,000. For additional information, call Kimball Mason, Vice President for Marketing at 617/237-2304.

FIRST MULTI-TASKING SYSTEM PERMITS 5 USERS ON SINGLE 'DEC PRO' SIMULTANEOUSLY

Advanced Computer Techniques Corporation (ACT) announced the first MUMPS multi-tasked operating system for Digital Equipment Corporation's (DEC) Professional series of microcomputers. The system is now available from Creative Socio-Medics Corp. (CSM), the medical division of ACT, through an exclusive U.S. distributorship agreement with the developer of the system, Structured Data Systems (SDS).

Called SMM300, the basic system supports up to three users. With the inclusion of a Real Time Interface board, a total of five simultaneous users can be supported. SMM300 is SDS' version of standard MUMPS. Over 100 users have obtained licenses for SMM300 since it was introduced in Europe about six months ago, according to Paul Stylos, SDS director, and president of the European MUMPS User Group.

SMM300 was developed to run on DEC's Professional 325 or 350 Personal Computer as a dedicated software system. It was designed for those who wish to run a multi-user based management system.

A single license for SMM300 costs $1,400. Quantity discounts are available, and inquiries from potential OEMs are invited. For details contact Gerry Fraenkel, Creative Socio-Medics Corporation, 16 East 32nd Street, New York, NY 10016 (212) 696-3500.

DEC SYSTEMS MADE AVAILABLE TO IRIS SOFTWARE USERS

PI Consulting Group, Inc. has announced the development of translation tools to convert IRIS Business BASIC to run on DEC PDP-11 and VAX computer systems. Designed for use on the target system, programs and procedures enable efficient conversion to RSX-11M, RSTS/E or VAX/VMS operating systems.

This development is expected to benefit software vendors as well as users. IRIS-based vendors seeking new markets have an extraordinary opportunity in DEC's popular mini and super-mini series. IRIS users can preserve their software investment while migrating to more powerful and universal systems.

Cost of the conversion will depend on the complexity of the application. For further information contact PI Consulting Group, Inc. 5150 E. Pacific Coast Highway, Long Beach, CA 90804, (213) 494-4702.

ADR/DATA RELEASED FOR THE 300 SERIES PERSONAL COMPUTER

Applied Data Research, Inc. announced today the availability of ADR/DATA, the first offering of its GENIE Series software for the DEC Professional 300 Series personal computer.

ADR/DATA, a relational data management system, is one of three integrated products for the DEC Professional. The others are ADR/GRAPH, a full-color business graphics package, and ADR/STATS a statistical analysis and forecasting product for business applications. All three products are completely menu-driven, providing a great deal of functional capability in an extremely easy-to-use environment. In addition to their integrated capabilities, all three products can be used as powerful standalone packages.

ADR/DATA, ADR/GRAPH, and ADR/STATS comprise The ADR GENIE Series. The GENIE Series products run on the complete line of DEC Professional hardware including the hard disk 350, the floppy disk 350, and floppy disk 325.

The GENIE Series software is being sold by ADR's Application Products Division. The products also are being submitted to Digital Equipment

To: David Holroyd, PRO-FPS Sales,
P.O. Box 19, 1 Redcliff Street,
Bristol BS99 7JS England.
Tel 011.44.272.24181
Telex: 449502 AMANDS

NAME

POSITION

COMPANY

ADDRESS

CITY STATE

ZIP TEL

You'll be amazed by the speed and power of PRO- FPS. It's a remarkable Decision Support System that far exceeds the capabilities of simple spread sheet analysis.

We are inviting you to send for the list of US Distributors where you can see for yourself how PRO-FPS really does make maximum use of the PROFESSIONAL 350.

With over 60 built-in financial routines, PRO-FPS gives you complete planning and modelling at your own desk. So it's a superb performer for corporate budgeting, project evaluation and cash planning.

Send for the list today and we are sure you will soon see why PRO-FPS is enjoying so much International success. Distributor enquiries are also welcome.

PRO-FPS

THE INTERNATIONALLY PROVEN FINANCIAL PLANNING AND MODELLING SYSTEM

NEW PRODUCTS

Corp.'s Digital Classified Softare (DCS) program for publication and distribution by DEC. ADR/DATA sells for $600, ADR/GRAPH for $300, and ADR/STATS for $500 with combination packages and quantity discounts available. Further information may be obtained from Applied Data Research, Inc., Route 206 & Orchard Rd., CN-8, Princeton, NJ 08540.

RSTS TO VAX MIGRATION PERFORMANCE TOOL

Responding with a solution to unexpected performance problems RSTS software users have encountered in the upward migration to VAX, Park Software has developed BIO, a RSTS style Disk I/O environment for native mode VAX. BIO enables the performance users were expecting when they upgraded to VAX without losing any of the VAX system advantages. This is possible because BIO focuses on the I/O problems rather than providing total RSTS emulation. For more information on BIO please contact Park Software, P.O. Box 202, Northgate Station, Seattle, Washington, 98125, (206) 284-4025. Demonstration copies are available and dealer inquiries are invited.

DEC COMPUTERS CAN NOW BE USED TO RUN INTEL SOFTWARE

Decmation of Santa Clara has announced a hardware-software package for DEC PDP-11 and LSI-11 computers which allows these minicomputers to be used to run Intel microprocessor development software which is normally run under the ISIS-II operating system on Intel MDS-800 and Intellec computer systems.

The package consists of an 8080 compatible processor which plugs into the DEC system and software which emulates the ISIS operating system. All the peripherals of the DEC system, such as disks and printers may be used. Data may be transferred to an Intel computer system by floppy disk or a serial communications link. The package costs about $2000 for a single-user system and less than $5,000 for a 3-user system. In most cases, the Intel software executes 2-3 times faster on the DEC system. CP/M is also supplied with the system, allowing the customer access to the many software tools available under CP/M. Contact Decmation, 3375 Scott Blvd., Ste. 422, Santa Clara, CA 95051, (408) 980-1678.

SMS ANNOUNCES 35 POUND DEC COMPATIBLE COMPUTER

SMS is now shipping the industry's most compact DEC table top microcomputer.

The MDX-11 is a complete LSI-11 16 bit microcomputer. It contains a DEC compatible Q-bus backplane which can be configured with the LSI-11/2 or LSI-11/23 CPU. The quad size backplane can accommodate up to 8 dual height Q-bus cards. The system can be confined with 22-bit memory for a total of one megabyte of memory.

A 5¼" Winchester disk is an integral part of the MDX-11 system. The non-removable disk gives the highest reliability available. The disk, in conjunction with the SMS FWD0106 disk controller, allows the maximum performance possible. The disk is formatted with 512 bytes per sector with no sector interleave for contiguous high speed DMA transfers. The MDX-11 can be configured with 10.6 or 15.9 megabytes of formatted data. Higher capacity Winchester disk drives will be available in the near future.

The MDX-11 contains a double sided/double density slim line floppy. The floppy disk is completely RX02 compatible. In addition, the controller allows automatic recognition of DEC floppy formats (RX01, RX02, RX03) and IBM floppy formats (3740 and 2/2D).

The MDX-11 can be supplied with complete RT-11 or RSX-11M distribution. General licenses for all DEC software and complete documentation are also available from SMS.

Price for the 10.6 megabyte version of the MDX-11 system without DEC CPU and memory is $5,700 at quantity 15. Price for the 10.6 megabyte version with LSI-11/23, 256 Kb of memory and DLV-11J is under $9000. Delivery is 45 days. For further information contact Michael Liccardo or David Weber, Scientific Micro Systems, Inc., 777 East Middlefield Road, Mountain View, California 94043, (415) 964-5700.

Compact 35 lb. DEC compatible computer from SMS.

PROFESSIONAL etc

NEW PRODUCTS

HIGH LEVEL PROGRAMMING SYSTEM NOW AVAILABLE FOR ADAGE 3000

Adage, Inc., a manufacturer of interactive computer work stations, and Intermetrics Incorporated, an international computer software and systems engineering company, today announced the availability of ICROSS-3000, a high level programming system developed by Intermetrics, for use with Adage 3000 Color Raster Systems on Digital Equipment microcomputers.

Styled after the C programming language, ICROSS-3000 supports compilation of microcode modules for the Adage 3000 bipolar BPS32 microprocessor. The package allows subsequent link-editing to produce load modules for microprocessors that can be downloaded directly after compilation. ICROSS-300 increases both programming productivity and maintainability at the compiler level, while retaining efficiency in execution time at the assembly level. For further information contact Adage, Inc., One Fortune Drive, Billerica, MA 01821, (617) 667-7070.

NEW OPERATING SOFTWARE FOR BUSINESS AND PERSONAL MICROCOMPUTERS

Microsoft unveiled today Microsoft Windows, an extension to their MS(TM)-DOS operating system that provides a universal operating environment for the new generation of bit-mapped application programs. Microsoft Windows will allow independent software vendors to develop sophisticated graphically-based integrated software packages that run without modification on any 16-bit microcomputer.

Microsoft Windows features a window management capability that allows a user to view unrelated application programs simultaneously. Further, it provides the capability to transfer data from one application program to another. Microsoft Windows provides these advanced user interface capabilities for owners and purchasers of mid-priced 16-bit computers.

Microsoft Windows will be sold like MS-DOS, adding little or no cost to a system. The hardware requirements for Microsoft Windows are; 192K bytes of random access memory, a mouse, two floppy disk drives and a bit-mapped display. For detailed information contact Microsoft Corp., 10700 Northup Way, Bellevue, WA 98004, (206) 828-8080.

NEW SECURITY/AUDIT TRAIL PROGRAM FOR VAX/VMS SYSTEMS

Innovative Software Solutions, Inc. has announced its newest product. Known as VAXGUARD, this program can be set to trap and record all interactive VAX/VMS DCL commands entered into your VAX/VMS system by one user, or, every user in the system, or, any number of users in between.

In addition to the command information, VAXGUARD also provides the system manager with the terminal that issued the command, along with a date/time stamp so you know exactly what happened, and when. Write ISS at P.O. Box 383, 67 Bridle Road, Billerica Mass., 01821.

SOFTWARE SUPPORT PACKAGE REDUCES DEVELOPMENT TIME

A new software package designed to ease the task of performing measurement and control functions with RTI-11-, RSX-11M- and RSX-11S-based systems is available from Analog Devices, Inc. The AC1815 software package consists of a loadable device driver and interface module which unburdens users of PDP-11- and LSI-11-based computers from having to write software to control Analog Devices' μMAC-4000 units. The μMAC-4000 is a single-board, intelligent measurement and control system which integrates sensor signal conditioning, linearization, analog-to-digital conversion, scaling to engineering units, digital I/O, and serial communication. These features allow users to easily hook up computers to analog signals.

The package is delivered as a RX01 diskette, or an RL01 or RL02 cartridge. It occupies 1.5K words of memory and is written in MACRO-11 for maximum efficiency. A commented source code is also included to allow modifications. Prices for the AC1815 start at $310 and it is deliverable from stock. For Sales/Engineering Information contact System Applications, Analog Devices, Inc., Two Technology Way, Norwood, MA 02062, (617) 329-4700.

BELDEN/CPD 386-G ELECTRONIC EQUIPMENT CONNECTOR

Belden/CPD has introduced a molded PVC connector for electronic equipment power supply cords which protects against accidental disconnects (Belden 386-G). The connector attaches to the power receptacle by utilizing full mounting ears and jack screws. It is designed to mate with Belden 17252, 17253, and 17265 or equivalent "international" power receptacles.

The 386-G connector features Belden's new box contact design, which prevents any molding compound from entering the contact area. This design also incorporates the "B" crimp on the contact to provide the best connection to the conductor. The results are superior electrical performance, low temperature rise, and optimum mechanical strength. These additional Belden features help eliminate equipment malfunctions due to faulty connections.

The Belden 386-G connector has a rating of 15 amps-125 volts/6 amps-250 volts. It is UL listed and

CSA certified. The standard color is black, and any color is available by special order. The cordage is available in two meter long shielded and unshielded versions. For additional information about the Belden 386-G electronic equipment connector, write to Manager, Marketing Communications, Belden/CPD, 2000 S. Batavia Avenue, Geneva, IL 60134.

COST-EFFECTIVE MAG-TAPE PERIPHERAL PROCESSOR SUPPORTS PDP-11 COMPUTERS

Computer Storage Technology (CST) announces a micro-programmed magnetic-tape peripheral processor which supports the Digital Equipment Corporation PDP-11 series of Unibus computers. By combining economy of design with reliable performance, it has become the most cost-effective unit of its kind available on the market today.

The new CST Model TC-200 is a single-board processor which takes only one SPC slot of the computer backplane instead of the two to four slots controllers usually require. It contains the logic necessary to control up to four 9-track tape drives at speeds ranging from 25IPS to 125IPS and can be used with NRZI (800 BPI), Phase Encoded (PE-1600 BPI), Dual Density (800/1600 BPI) or any combination of the three in either DEC or IBM standard packing formats.

The TC-200 supports all DEC operating systems which include software drivers for RSTS/E, RSX-11M and RT-11. Diagnostics are executed in the standard ZJF12-RB kit. No software driver or diagnostic changes are required. Priced at $3,000 for a single board, the TC-200 is designed for minimum component count with low manufacturing cost and high reliability. Delivery is 30 days ARO. Quantity discounts are available. For more information contact Computer Storage Technology, 1369 S. State College Blvd., Anaheim, CA 92806, (714) 778-3656.

NEW VERSAPLOT SOFTWARE SUPPORTS ELECTROSTATIC COLOR PLOTTING

Versaplot Color Random software from Versatec, a Xerox company, is the first plotting software to support electrostatic plotting in color. The new package provides for eight line colors and 256 pre-defined area colors. An additional 256 colors may be defined by the user. The software also supports black and white plotting.

CalComp-compatible application programs can be used without modifying code. Menus for plot, plotter, viewport and color options enable user control of various run-time parameters. Versaplot Color Random output can be directed to on-line plotters, mag tape, or remote devices. User may define plotter speed, request multiple copies, and change defaults at run-time.

The software is integrated to run under DEC VAX/VMS and IBM OS/CMS operating systems. "Universal" packages are available for use with other computers and operating systems.

VAX/VMS Versaplot Color Random is available on RX01 diskette, nine-track mag tape (800 or 1600 bpi), and TU58 cassette for $4,000. Universal packages, ASCII or EBCDIC nine-track mag tape (800 or 1600 bpi), are $2,500. Users of earlier monochrome Versaplot may upgrade to Color Random for the price difference between the two license fees plus $500 and proof of license. For more information, write for "Versaplot Color Random" from Versatec, a Xerox company, 2710 Walsh Avenue, Santa Clara, California 95051, or call (408) 988-2800.

Versatec Versaplot™ Color Random software — the first plotting software to support electrostatic plotting in color.

PROFESSIONAL ^etc

PRODUCT UPDATES

POPULAR DBMS FOR VAX ENHANCED

Significant efficiency increases — 80% to 95% in certain cases — are achieved by the newly released version of System 1032 Data Base Management System for DEC VAX computers.

New programming features, handling of EBCDIC data, and a simpler PASCAL interface are also included in Version 1.5. System 1032 is produced and licensed by Software House, Cambridge, Massachusetts.

In contrast to other DBMS products, System 1032 is written in MACRO exclusively for the VAX.

System 1032 operates on all VAX models, using the popular relational approach to data management. The system is the outgrowth of over ten years of Software House experience in developing commercial DBMS products for the DEC environment. Software House offers a 60-day full system trial for $85. A typical System 1032 license costs $40,000 for the VAX 11/780. For more information, contact Betsy Ziegler, Software House, 1105 Massachusetts Avenue, Cambridge,MA 02138,(617) 661-9440.

ANNOUNCIING ARSAP FOR RSX-11M VERSION 4.1

ARSAP, the Resource Management and Chargeback System for Digital Equipment Corporation computers, has been updated to run on PDP-11 computers using the recently released RSX-11M 4.1 operating system.

ARSAP is a comprehensive computer resource management system which provides users, system managers, corporate and financial staff with reliable reports for making informed and accurate decisions about the data center operation.

ARSAP is available on all of the most widely used DEC operating systems: RSTS, RSX-11M, RSX-11M-PLUS and VAX/VMS. Managers of multiple operating system shops can get compatible reporting of resource utilization, making it easy to compare productivity on all in-house DEC systems. This compatibility is not available from any other vendor, including DEC.

ARSAP for RSX-11M can be licensed for $2995 and multiple copy discounts are available. ARSAP is delivered within two weeks, comes with a 30-day acceptance period, a full one year warranty, including maintenance and enhancements services, and complete documentation.

Existing ARSAP for RSX-11M customers who are under the ARSAP Maintenance and Enhancement Service are eligible to receive the updated version of ARSAP at no charge. ARSAP customers not under the service can purchase an update for $1,198. The update comes with documentation and one full year of support. For more information contact GEJAC, Inc., P.O. Box 188, Riverdale, MD 20737, (310) 864-3700.

ACCESS CONTROL DEVICE FOR COMPUTER NETWORK SECURITY

Staff announces the immediate availability of a new version of THE KEY for use in controlling remote access to a computer system. THE KEY easily attaches to the serial port of any RS-232 terminal and normally requires no other modification to the hardware.

THE KEY extends security well beyond that level which would be obtainable by using software techniques alone or by using a triggered fixed identification response. THE KEY achieves its level of security by being secure against both duplicatic and simulation.

Security against duplication of TH KEY is assured by using an integrate circuit which requires expensive reverse engineering to duplicate.

Security against simulation is obtained by having THE KEY respon to each of a number of inquiries with a response which not only depends on the inquiry, but also on the specific key being queried. The number of unique inquiries, responses, and versions of THE KEY is about 24 quintillion (24 with 18 zeroes after it) for each.

For additional information about THE KEY, please contact Mary Gibson, Staff Computer Technology Corporation, 10457 J Roselle Street, San Diego, CA 92121, (619) 453-0303.

'WATCHDOG TIMER' ADDED TO MULTI-PURPOSE Q-BUS BOARD

Codar Technology, Inc., today announced a new version of its Q-Timer single-board system module that adds a watchdog timer to existing functions, such as calendar clock and non-volatile CMOS memory.

Designed for integrators of real-time systems based on Digital Equipment Corporation's family of microprocessors, the dual-wide Q-Timer board is the only product of its kind with complete software support stored in an on-board EPROM, according to Doug Harts, marketing manager at Codar's Computer Products Division.

The board provides built-in compatibility with DEC's 16, 18 and 22-bit bus architectures and, in some instances, can replace up to four dual-wide modules.

The watchdog timer makes the Q-Timer particularly suitable for

unattended or remote operations in that it will re-boot a system if program execution stops because of a non-catastrophic problem.

Expandable 8-bit parallel input and output ports provide a path by which the Q-Timer can perform a variety of I/O functions in addition to routine calendar clock tasks, often eliminating a DRV11-type interface board. For further information contact Codar Technology, Inc., 1428 Florida Ave., Longmont, CO 80501, (303) 776-0472.

MTLIB: LIBRARY OF ROUTINES

MTLIB from NAB Software Services, Inc., Albuquerque, New Mexico, is a library of routines for a General Purpose Magnetic Tape Driver. These routines permit an application program to assume complete control of the magnetic tape unit and to make conversions between ASCII and EBCDIC. Thus, an application program may read or write a tape in any arbitrary format. MTLIB also includes the executable program, MTAPE, that provides pseudo keyboard or monitor commands for initiating certain magnetic tape operations. The routines and program in MTLIB are usable under both RT-11 and TSX-Plus with either the MM (TJU16), MS (TS11), and the MT (TM11) tape handlers.

The library includes the tape driver subroutines, the ASCII-EBCDIC conversion routines, and a number of language interface routines. The language interface routines permit calling the tape driver and the conversion routines from programs written in the "C" language, COBOL-Plus, DBL, FLECS, FORTRAN, MACRO-11, PASCAL, and RATFOR.

The binary kit contains the implemented library file, the MTAPE program, the user's guide document in machine readable form, and the sources for tape driver demonstration programs in each of the supported languages. Contact NAB Software Services, Inc., for more information, 3236 Candlelight Dr., NE, P.O. Box 20009, Albuquerque, NM 87154, (505) 298-2346.

TWO ADDITIONS ANNOUNCED FOR PATHWAY SERIES

Technology Concepts Inc. announced today the addition of two new products to its Pathway series of

gateway products that allow DECnet operation over ring and broadband based local area networks.

The fully compatible and migratable products called RINGWAY and BROADWAY allow users to interface DECnet to network data link and physical link layers not supported by Digital Equipment Corporation.

Pathway products make the LAN channels appear as standard DECnet links; require no DECnet software code modification; provide full DECnet function: file access, file transfer, remote terminal access, and task-to-task communication; support full routing for large networks with one or more Pathway products and DECnet point-to-point links; and simultaneously supports non-DECnet VAX and RSX applications directly to the LAN channel. Additional information is available from Technology Concepts Inc., Old County Road, Sudbury, MA 01776, (617) 443-4637 or 443-7311.

IMCSmicro RELEASED FOR VAX, PDP-11 MICRO AND SYSTEMS COMPUTERS

Interactive Information Systems of Cincinnati, Ohio announces release of IMCSmicro. IMCSmicro offers functionality comparable to that of IMCS, the interactive manufacturing resource planning and control system, presently operating on Digital Equipment Corporation's VAX hardware with over 90 installations. IMCSmicro will operate on Digital's PDP11 micro and systems computers.

IMCSmicro is an online, interactive system flexible enough to meet the needs of job shop, repetitive and process manufacturing. IMCS is presently operating in many small as well as very large companies. The design of the system supports a modular approach as required to run your business. Incorporated into the system is an easy to use, friendly report generator for ad hoc reports. IMCSmicro comes complete with documentation. Complete courses for application and technical training are available. For further information contact David Lang, CPIM; Director of Marketing, Interactive Information Systems, 10 Knollcrest Drive, Cincinnati, OH 45237, (513) 761-0132.

NEW PRICING SCALE FOR ROSS/V SOFTWARE

Evans Griffiths & Hart announces a new pricing scale for its ROSS/V

PRODUCT UPDATES

software package. The RSTS/E Operating System Simulator under VMS will continue to be offered at $10,900 for a single-CPU VAX-11/780 license. It is being newly offered at $7,700 for a VAX-11/750, and at $5,500 for a VAX-11/730.

Written in VAX-11 MACRO, ROSS/V provides the fastest way to bring up RSTS/E applications on the VAX, and the only way to do RSTS/E development on the VAX. ROSS/V supports an extensive subset of RSTS/E monitor calls, standard RSTS/E features like CCLs, DOS-formatted magtape, and RSTS/E-style update mode, the "hidden" RSX run-time-system (with 32 KW job size), resident libraries, job spawning and detached jobs, spooling to VMS print and batch queues, and mailbox send/receive for communication with VAX-11 BASIC and other native mode applications.

OEM and quantity discounts are available. Contact Evans Griffiths & Hart, Inc., 55 Waltham Street, Lexington, MA 02173, (617) 861-0670.

MANUFACTURING QC OPTION INTRODUCED FOR RS/1 SOFTWARE USERS

A manufacturing quality control option for users of RS/1 — The Research System software has been introduced by BBN Research Systems division of Bolt Beranek and Newman Inc. RS/1 runs on Digital Equipment Corporation's VAX and PDP-11 minicomputers.

Called the RS/1 Quality Control Analysis Option, the graphics, statistics, and analysis package aids design of manufacturing quality control procedures when used with the RS/1 software. The quality control analysis routines take data in RS/1 tables as input, and output RS/1 tables and graphs. They use interactive dialogues to solicit information from the user, thus requiring little training before use.

RS/1 is used by scientists and engineers in a wide range of industries and applications, including pharmaceutical, chemical, electronics, environmental monitoring, consumer products, genetic engineering, and general manufacturing. It offers a full range of data-handling techniques: data entry and storage, two- and three-dimensional graphics, curve fitting, statistical analysis, and analytical modeling. For further information contact BBN Research Systems, Div. of Bolt Beranck and Newman, Inc., 10 Moulton St., Cambridge, MA 02238, (617) 491-8488.

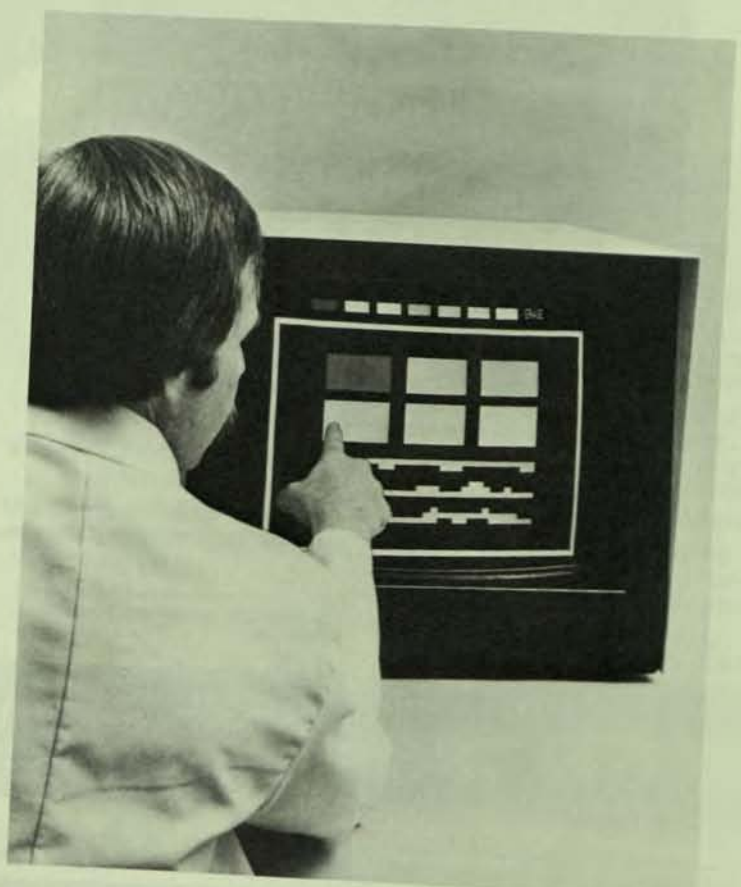
PRODUCT UPDATES

CARROLL UNVEILS INFRARED TOUCH INPUT SYSTEM

An infrared touch input system from Carroll Touch Technology makes the quick, easy touch interface available for the popular, low-cost ISC 8000 series of 19″ industrial color graphics terminals. The touch system is expected to find wide acceptance in process control, factory data collection, and shop floor control applications.

Carroll's touch system uses LED emitters and phototransistor detectors mounted around the video screen, creating a lattice of infrared light beams just in front of the display surface. When the screen is touched, light beams are interrupted, and the computer responds to the reported X-Y coordinates of the touch. No typing skill or computer experience is needed to operate the touch system.

Unlike membrane touch systems, the Carroll system has no overlay between the viewer and the screen, leaving screen brightness and resolution unaffected. The LEDs and phototransistors are hidden inside the terminal bezel, so that a touch terminal looks exactly like a conventional terminal. In addition to being natural and easy to use, the touch input system provides a computer interface that is both fast and accurate. For information, write or call Carroll Touch Technology, 2902 Farber Drive, Champaign, Illinois 61821, (217) 351-1700.



An infrared touch input system from Carroll Touch Technology is now available for the popular, low-cost ISC 8000 series of 19″ industrial color graphics terminals.

PASSPORT FROM DATALEX OFFER FULL PROTOCOL FILE TRANSFER

The Datalex Company has announced the release of a newly enhanced version of Passport, its highly adaptable remote communications program. Passport is the only single piece of software available that provides full protocol file and program transfer from micro to micro and between micro and remote computer.

Passport is a bi-directional communications package that simulates synchronous protocol on inexpensive asynchronous lines. It is designed for use in any circumstance where microcomputers, linked to a remote computer, are used in distributed data processing. DDP operators may now communicate micro to micro as well as micro to remote over asynchronous lines through an asynchronous port.

Passport will transfer both ASCII and Binary files. Binary transmission can be useful for distributing programs from a central location to ensure uniformity of applications, and since Passport may be driven from either end, unattended uploading from PC to remote is possible. Passport also provides ASCII terminal emulation.

Micros using Passport include the IBM-PC and XT, DEC Rainbow, Compag, Sage and the H-P 125. Passport also operates on a number of remote computers including IBM/CMS, DEC-10 and DEC-20, DEC VAX, H-P 3000, and Prime 400 Series. Passport can be adapted to virtually any other remote computer by using Datalex's Remote Implementation Kit, which sells for $100.

Passport is available in OEM quantities 50-100 at $146 per unit, with pricing available for end users and small quantities as well. A special 10-micro, 1 remote package is available off the shelf priced at $4450. For more information contact The Datalex Company, 650 Fifth Street, San Francisco, CA 94107, (415) 541-0780.

PROFESSIONAL etc

PEOPLE, PLACES, THINGS

VAX SOFTWARE COMPANY ESTABLISHED

One of Digital's largest West Coast system houses has formed a new subsidiary to meet the ever increasing demand to economically license VAX software. Lantor Information Systems has begun by supplying users of DEC computers with an affordable means to license the Operating System, Data/Forms Management Software and the Language Compiling Programs. The firm trades exclusively in the VAX/VMS environment. Many companies find the expensive "A" license to be more than they need. They often merely require the right to use the software and obtain distributions. A "DZ" license provides such companies with savings of about 50% over the primary license fee.

Lantor currently offers single user licenses and binary distribution code, complete with documentation, for a wide variety of Digital software. They offer the VMS Operating System and Data Management Software including CDD, DBMS, TDMS, and DATATRIEVE. They also offer the FMS, ADE, and CMS Utility Software. In addition they have all the major languages for VAX; VAX-11 BASIC, BLISS, "C", COBOL, CORAL-66, DIBOL, DMS, FORTRAN, PASCAL and PL/I. They supply Communications Software such as DECnet, PSI, and the I.B.M. protocol emulators. And they also handle several Digital applications. These include DECalc, DECmail, DECtype, DECspell, DECgraph, DECslide and ReGIS.

All of these licenses have the option of upgrade and full support from Digital. For further information on the products and pricing please contact Matthew Owen, Lantor Information Systems, 8055 Manchester Avenue., Playa Del Rey, CA 90291, (213) 821-0642.

DEC SIGNS SALES ACCORD WITH ACCOUNTING SOFTWARE FIRM

In a unique product distribution contract, Digital Equipment Corporation has agreed to provide field sales support for a multi-journal accounting software package produced by Prodata, Inc., of Albuquerque, New Mexico. Prodata President Thomas Haney announced the agreement, noting that the MJA package is "one of a very limited number of products sponsored by DEC in the field." DEC provides support for approved software through the Digital Certified Software, or DCS, process.

MJA provides General Ledger, Accounts Payable, Accounts Receivable, Payroll/Personnel and Order Entry/Inventory journals that function independently, or as a fully integrated system. It's user-friendly, table-based and menu-prompting, eliminating the need for a programmer.

Digital Equipment Corporation will conduct comprehensive training seminars for DEC sales people to familiarize them with features of the accounting package. MJA will be made available through direct field sales, as well as in Digital Business Systems Centers. For additional information about the MJA accounting package, contact Thomas Haney at Prodata, Inc., 12101 Menaul Blvd. N.E., Albuquerque, N.M. 87112, (505) 294-1530.

IDS ANNOUNCES PYRAMID CONTRACT & DEC TRADE-IN & EXCHANGE PROGRAM

International Data Services, Inc. (IDS), a Santa Clara-based UNIX provider of state-of-the-art software development tools, has been named as a distributor for the Pyramid 90x system in the Northern California area.

The $4 million contract signed with Pyramid Technology Corp. of Mountain View calls for IDS to market and support the Pyramid 90x, a UNIX-based, 32 bit virtual memory super-minicomputer, to the end user. IDS expects that the major demand for the Pyramid 90x will be for use in the software development, engineering, scientific and R&D fields.

To promote the Pyramid 90x, IDS has announced the Digital TIE* LINE (Trade in and Exchange Program). This program is oriented towards owners of DEC PDP 11 and VAX minicomputers who have good, maintainable DEC equipment but require additional processor performance.

The exchange program offers customers a higher performance super-minicomputer, such as the Pyramid 90x system, in trade for their DEC minicomputers. For more information about the Pyramid 90x system and the IDS Digital TIE* LINE call IDS at (408) 986-1972.

DEC TO MARKET RSSP's PIOS SOFTWARE

Rath and Strong Systems Products, Inc. (RSSP) announced a joint marketing agreement with Digital Equipment Corporation (DEC) that

PEOPLE, PLACES, THINGS

will enable DEC to market RSSP's PIOS software system as part of their total hardware and software solution. In addition, DEC has selected PIOS to be utilized by their own manufacturing facilities in the United States and Western Europe. To date DEC has installed PIOS in five U.S. and two European facilities.

This agreement means that Digital and RSSP can now supply a wide range of various manufacturing companies with a package that provides functionality normally available on more expensive mainframe computers. PIOS software is now available for Digital's VAX Super-minicomputer. Information is available from Rath and Strong Systems Products, Inc., 14901 Quorum Drive, Ste. 600 , Dallas, TX 75240, (214) 980-0647.

DIGITAL DEALERS ASSOC. ELECTS OFFICERS

The Digital Dealers Association Fall 1983 Convention was held in October, in Scottsdale, Arizona at the Camelback Inn Resort and Golf Club.

New Officers and Directors elected for 1983-84 were as follows:

C.D. Smith, C.D. Smith & Associates, Inc., *President;* Allan Newman, Newman Computer Exchange, Inc., *Secretary;* Julian Sandler, Brookvale Computer Associates, *Treasurer;* Jim Claypool, American Computer Group, Inc., *Board Member;* Frank Fusco, One Stop Computer Shop, Inc., *Board Member;* Joe Corrone, Newcorp Products, Inc., *Board Member;* Bryan Eustace, Thomas Business Systems, Inc., *Board Member.*

New members elected to Digital Dealers Association were Jack Clinton of Clinton Digital, Inc., and Ray Clairmont of QEI, Inc.

Digital Dealers Association (DDA) was founded in March of 1982, its primary objective being the promotion of an orderly secondary market for the products of Digital Equipment Corporation . Among the activities projected to achieve this objective are:

1. Establishment of a dialogue with Digital Equipment Corporation to encourage the adoption and enforcement by DEC of consistent policies with respect to various matters.

2. Establishment and enforcement of standard of professional practices and conduct for members of the Association for the protection of other members and end users.

3. Education of end users and other dealers with respect to the benefits of doing business with members of the Association.

The Digital Dealers Association, a Michigan non-profit corporation, may be reached at P.O. Box 8140, Ann Arbor, Michigan 48107.

DSD EXPANDS SAN JOSE FACILITIES

Data Systems Design, San Jose, has announced the move of its engineering and marketing departments into 30,000 square feet of space at 2125 Lundy Avenue, San Jose. The expansion, into a building adjacent to existing headquarters at 2241 Lundy, represents a 50 percent increase in facility space for the 9-year-old manufacturer of disk storage systems and controllers.

DSD's last major facility expansion was in 1980, when the company moved into the 2241 Lundy location. With the new facility, DSD's manufacturing and corporate headquarters space now totals 94,000 square feet.

The mailing address and telephone number for the engineering and marketing departments remain 2241 Lundy Avenue, San Jose, CA 95131, (408) 946-5800

DIGITAL, PHOENIX DATA SYSTEMS SIGN COOPERATIVE MARKETING AGREEMENT

Digital Equipment Corporation and Phoenix Data Systems announced an arrangement to market cooperatively a software package for verification of very-large-scale-integration (VLSI) mask sets that will run on Digital's VAX 32-bit superminicomputers. The package, called MASKAP, is used in integrated-circuit design verification prior to silicon fabrication.

According to provisions of the agreement, representatives from Digital and Phoenix Data Systems can conduct joint visits to customer sites to provide information on their companies' products.

CARE LAUNCHES AGGRESSIVE DISTRIBUTOR PROGRAM WITH UCC DIGITAL SYSTEMS

Care Informations Systems' President John Struckhoff announced the start of an aggressive new distributor program with the signing of an agreement for UCC Digital Systems of Florida to market the CARE/DM System medical and dental software.

The agreement specifies that UCC Digital will sell a minimum of 500 DEC-compatible CARE/DM System packages over a specified period. End-user prices range from $15,000 for the PDP 11/23 to $35,000 for the VAX 11/780.

The CARE/DM System is a computerized version of the common pegboard manual management system used by physicians and dentists which makes it easy for users to convert to the automated system.

Care is currently in the process of selecting other qualified distributors. They offer a complete marketing package consisting of training and sales video tapes, sales training classes, cooperative advertising. For further information or for a distributor application, contact Care Information Systems, Inc., 3009 South 6th Street, Springfield, IL 62703. (217) 522-CARE

Send Classified Ads to: Classified News, c/o THE DEC PROFESSIONAL, P.O. Box 362, Ambler, PA 19002-0362.

Classified ads are priced at $1.00 per word. Display ads are $75.00 per column inch. If we set, this includes border and 2 lines in bolder and/or larger type size, if desired, — please specify.

GENERAL LEDGER SYSTEM FOR VAX

Have problems getting timely reports from your accounting system? Need better control over your financial reporting? Use PLYCOM's General Ledger Package for accounting software that is easy to use, yet effective. Includes excellent support, complete documentation and:

- Easy to use menus
- Excellent audit trail
- User defined ledger codes
- Budget reporting
- Flexible consolidations
- User defined financial reports
- Multiple financial statements
- On-line account inquiry
- Interfaces to Payables, Receivables and Fixed Assets Packages
- For VAX/VMS

Plycom services, inc.
P.O. Box 380465
San Antonio, TX 78280
(512) 734-4366

RSTS RESCUE SQUAD

We salvage all kinds of disasters:

- unreadable disks
- ruined UFDs and MFDs repaired
- immediate response
- telephone DIAL-UP
- on-site
- software tools
- custom recovery
- 90% success to date
- more than 2 GB rescued to date

Brought to you by
On Track Systems, Inc.
and a well known (and read)
RSTS expert.
CALL 24 HOURS
215—542-7008

ASSEMBLY TRANSLATION

• Macro-11 to MC68000 •
Cross-Assembly, Macro-11 Style
• MC68000, MC68xx, Z800x, Z80 •
with
Structured Assembly
• MC68000, Z800x, MC6809 •
for
RT, TSX, TSX+, RSX, UNIX, VMS

SSG
Box 1595
North Wales, PA 19454
(215) 699-9658

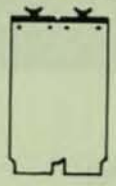
BUY SELL TRADE

DEC PDP-11

SYSTEMS
PERIPHERALS
COMPONENTS

dce DIGITAL COMPUTER EXCHANGE INC
27773 Industrial Blvd., Hayward, CA 94545
CALL-(415) 887-3100

BUS GRANT CARDS



Dual wide, with handles
specify Q-BUS or UNIBUS
$24.95 each
Immediate Delivery

Scientific Computer Systems, Inc.
2438 30th Street
Boulder, Colorado 80301
303 447-0353

FOR SALE: VAX 780 memory.
5 Mostek 256K boards (1.25MB total)-$1500; 4 National Semiconductor 256K boards (1.00MB total)-$1500; 4 Standard Memories 256K boards (1.00MB total)-$1500.
Also, NEC Spinwriter model 5520-$1500.
Contact: Stuart Koop, Millikin University, 1184 W. Main, Decatur, IL 62522, (217) 875-4098.

VAX AND PDP DEVELOPMENT TIME

NO KILOCORE TICK CHARGES / NO CPU CHARGES

Omnicomputer™

$7/$14
RSTS/E VMS

PER HOUR CONNECT TIME

BUDGET BYTES™ 212-944-9230

Omnicomputer, Inc.
1430 Broadway, New York, N.Y. 10018

VAX SALE

Your Source For DEC CPU's

VAX 750	PDP 11/44
11/23+	MICRO-11
RA 81	TU 80
CDC	EMULEX

(714) 951-7769
P.O. BOX 2000-21 MISSION VIEJO, CA 92690

Digitaldata Inc.

Computer Hardware Engineer

Temple University has an immediate opening for a Computer Hardware Engineer. The qualified candidate will have at least 3 years experience in PDP-11 computer repair and expertise in the design and debugging of both linear and digital circuits. This position is responsible for the maintenance of computer equipment, administrative duties, and design and building of support hardware. Please send resume with salary requirements to: PERSONNEL SERVICES, Rm. 203, 1601 N. Broad St., Phila., PA 19122.

TEMPLE UNIVERSITY
Equal Opportunity Employer

JOB OPPORTUNITIES — COUNTRY WIDE & OVERSEAS

VAX • RSX-11 • PASCAL • BASIC • MESSAGE SWITCH
SOFTWARE ENGINEERS • SYSTEMS • SOFTWARE DEVELOPMENT

ENGINEER ONE is an Internationally recognized Employment Service specializing in Engineering, Management & Software Professionals. If you have 1 to 12 years experience in any of the above functions, send your resume ASAP or call Lee Skinner @ 800-251-9847 or 615-690-2611.

ENGINEER ONE INC.; P.O. Box 23037; Knoxville, TN 37933

LIST OF ADVERTISERS

A Masterpiece in design, performance and reliability...

with service to match.

Plessey, an accomplished master in the computer and electronics industry, manufactures a broad line of DEC* - compatible products. With over fourteen years of experience, Plessey is firmly committed to producing quality products you can rely on.

Plessey memories use the latest technology. We offer one megabyte memories for the VAX*, LSI* BUS and UNIBUS* systems. Our LSI 11/23 and UNIBUS memories support 22-bit addressing and give you a system compatibility of up to four megabytes of main memory. Every memory is thoroughly tested and comes with a one-year, return-to-factory warranty.

Plessey also manufactures a large variety of DEC - compatible controllers and interfaces for LSI and UNIBUS systems. Our one board controllers use less space and run cooler for greater reliability. The result is lower maintenance costs.

All Plessey products are sold at prices that make Plessey your best bargain — for single unit buys and large quantity orders. OEM, GSA and educational discounts are also available.

We offer direct, worldwide service from experienced, factory trained Plessey specialists. You get fast, responsive professional service from people who know their products — from the inside out.

To find out how you can add Plessey reliability to your system contact:
Plessey Peripheral Systems, Computer Products Division, 1674 McGaw Avenue, Irvine, CA 92714.
(714) 540-9945 or (800) 992-8744

PLESSEY. ASK ANYBODY.

CIRCLE D202 ON READER CARD

*DEC, VAX, LSI and UNIBUS are trademarks of Digital Equipment Corporation.

DEC Owners— HDLC for High Speed, Error Free Links.

HDLC is a powerful protocol designed to support high speed communication with sophisticated error checking. With ACC's IF-11/HDLC, DEC computers can now communicate at line speeds up to 300 kbps with other systems that support the HDLC industry standard. This frees you from proprietary protocols and increases your system design flexibility.

For high speed, error free, point-to-point links, call ACC (805) 963-9431.

We Make Advanced Computers Communicate.

ACC

Advanced Computer Communications

720 Santa Barbara Street
Santa Barbara, CA 93101
(805) 963-9431

CIRCLE D110 ON READER CARD

TAPE DIMENSION III

DEC COMPATIBLE CONTROLLERS

Tape Dimension III is the only buffered tri-density (GCR/PE/NRZI) TS-II™-emulating controller on the market. The combination of its unique asynchronous handshake design and 64K byte buffer enables it to take full advantage of bus speeds without the risk of causing data late conditions in other bus transfer operations. It makes Tape Dimension III particularly adaptable to systems with high speed disk drives.

The 64K byte buffer provides total immunity to data late conditions, even at high-speed data rates on a highly populated peripheral bus. In fact, Tape Dimension III actually has greater capability than the TS-II!

Tape Dimension III is completely software-transparent to the VAX™ and PDP-II™ Unibus environment including diagnostics in VMS.

The Tape Dimension III controller supports up to four dual-density (Pertec compatible) drives or four tri-density (STC or TELEX type) drives. It is a single embedded hex PC board that fits into any standard SPC slot.

This new dimension in tape transfer dramatically reduces the ratio of protocol to information data. And tremendously increases the efficiency of Unibus utilization. Find out how much it can improve your system throughput. Call or write today for complete information. WESTERN PERIPHERALS, Div. of WESPERCORP, 14321 New Myford Rd., Tustin, GA 92680, Tel: (714) 730-6250, Cable WESPER, TWX 910-595-1775, Telex 472 0629. Wesper International GmbH, Tel: 089 982420. Wesper International (U.K.), Tel: (44) 0276-20934.

TS-II, VAX and PDP-11 are trademarks of Digital Equipment Corporation

western peripherals

Division of WESPERCORP

CIRCLE D109 ON READER CARD