



THE NEWSLETTER FOR  COMPUTER USERS

SOMETHING WE CAN CALL OUR OWN

Logo by: R. Gerry Moore
Story by: Bruce N. Warner

Have you noticed there's a new masthead for the MOTD? How about that new banner across the front page? And look at this! Our own logo! Now we're talking!

A lot of effort has gone into the new MOTD. We have our own logo now, and we're growing beyond even your wildest expectations. At last count, we had somewhere around 1300 members, and we look like it.

When Gerry Moore, of Dale City, VA, was asked to come up with something that would be simple yet effective to portray the OS-9 Users Group, we gave him an open hand. He decided that it had to say something about computers. It had

to be clean. It had to be different. It had to be noticeable.

What he came up with was the design you see on this page. It's clean, simple and effective. It's uniquely our own and you'll see it everywhere we go.

Some will ask why we didn't just use the Microware logo. Simple, that's a trademark. Use of their logo would be the same as pirating software or publishing someone else's book under your name. It just doesn't work that way.

If you think Gerry's name is familiar, it's only because it is. Gerry's the same guy that designed the banner for the MOTD. I'm glad to have the opportunity to thank him here in the MOTD.

Enjoy having pride in the new UG logo, just like you have pride in the OS-9 operating system.



STRIP, ONE POWERFUL UTILITY

This description of the STRIP utility, along with the accompanying source code listing are from the OS-9 UG Library (disk #25). Thanks to Darryl Hock and Dave Kaleita, this program will help you transfer those files you have to do while on the road into your OS-9 system. STRIP is used to strip, add or process any combination of CRs, LFs, back spaces (processed) and/or control characters.

STRIP is a very versatile control character-stripping filter. Its default options clear the parity bit (bit 8) of every byte in an input file, and strip all control characters except carriage returns and linefeeds from the remaining characters. You can, however, select the "-L" option, in which case linefeeds will also be removed, and/or "-C", which will strip all carriage returns. Alternately, the "+L" option causes STRIP to add line feeds

after every carriage return in the file (even if linefeeds already exist). STRIP will also process all backspace characters in the file by selecting the "+B" option. When that option is used, a non-backspace character is also removed for every backspace character that appears in the processed file. This is useful when processing data saved from a session on a BBS.

In all cases, STRIP prints the number of characters stripped in a message to the standard error output when it is done. If you would also like to see which characters are being stripped during processing, select the "+D" option.

Note that options are processed in the order they appear on the calling line. For example, the line "Strip +b -l +d +l -l <InFile>OutFile" will strip all linefeeds and control characters appearing in InFile, process all backspaces, and print

all characters it removed so far to path #2. It would then add linefeeds after each carriage return in the file, and then remove them again! The "number of characters stripped" message would reflect the total removed only and not those added with the "+l."

Also note that the "+d" option did not print all characters stripped from the input file, but just those stripped by the options appearing before it on the calling line. As another example, use "Strip +L -C" to replace all carriage returns in a file with line feed characters. STRIP prints a help message if you type "Strip -?".

Since STRIP is written in assembly language, it doesn't require any additional software. It will run on all levels of OS-9 and all versions. It was originally written on a Hazelwood Helix.

```

00001      nam  Strip
00002      use  /h0/defs/defsfile
00003  0001      LEVEL      equ  1          select level one
00004      ifpl
00009      endc
00010      ttl  Filter to strip control characters from a

```

Continued to page 4

BULK RATE
U.S. POSTAGE
PAID
PERMIT NO. 18
PROSPECT, KY

86/04/17

GETTING IT TOGETHER!!!

The development of this issue. . .

by Bruce N. Warner, Editor

There's never a dull moment when you're in the OS-9 Users Group. Traveling from one end of the house to the other can be a journey through a lost galaxy. Some Hollywood producers would call it "a galaxy far far away." My wife just asks how I ever get in and out of the computer room. Last night it took 45 minutes to make the 15 foot walk from computer room to bedroom (yes, UG officers do sleep, but not much). A call from Brian, then one from Dale, then another from Fred. I finally decided to let the answering machine take the calls and ended up missing one from my parents.

A real improvement and soething new

The latest news is that both our new banner and our new masthead have received wide acceptance. And to make things even better, we're now ready to show off our new logo. It's been in the works for months now. The UG officers and directors accepted the design during a late night meeting at the Palo Alto RAINBOWfest. The design is by Gerry Moore. Gerry isn't into computers at all. He's an artist from Dale City, Virginia who works in Washington, DC, and one of my best friends. When the idea of a logo first came up I mentioned Gerry does that kind of stuff, and the UG officers decided to give him a chance. When we showed off his design at the Palo Alto RAINBOWfest it was an instant success. Even the people that just stopped by the booth thought it was a winner.

Speaking of Palo Alto, you wouldn't believe how well our booth did. I felt guilty being there. There's something about being able to share your knowledge with people that really want to know how to use our favorite operating system. The thing that really made me feel good was how many people wanted the MOTD bad enough that they were looking to buy a subscription, they were shocked to find out that it's free when you join the UG. That's only the beginning! There were at least ten people that came up to ask a question and ended up joining the UG because they felt part of the UG

from the way they were handled. One of them mentioned he felt the answers he received were worth the price of membership for the next ten years. Thanks to people like Brian, Bill, Steve and Dale.

Some goofs

Okay, so I'm not perfect, but neither is anyone else, and I'm far from the exception (just ask my wife). We're just now getting a feel for how much copy it takes to fill an issue of the MOTD, so I had made an assumption that there were certain articles that would fit in the last issue of the MOTD that didn't make it. I can start with the UG library reviews by Bert Schneider.

This takes us back to that "Inner Circle" I spoke of in the last issue. If there is anyone that should have his/her articles included because of their closeness to the UG officers, it's Bert. He has devoted more time and effort to the UG and our library than anyone I know. There's always someone behind the scenes, and Bert is the man behind the scenes of the library. I apologize publicly and personally to Bert for the missing review of the UG library disks. You'll find them in this issue.

Speaking of Bert Schneider, he's agreed to start a column that provides a mini-tutorial on an individual program from the UG library. What we'll try to do is have Bert select the program of the month and make his tutorial on that program. This will kill two birds with one stone.

A chat on Delphi

I had the pleasure of going on-line at the end of February for a DELPHI conference. The notorious Marty Goodman was out there stirring things up as usual (I may regret telling Marty we need people to stir things up on occasion), but he added a lot to my pleasure of being the guest speaker.

One of the nicest things to happen was my request for people to write for the MOTD. I thought it was a good idea until tonight when George called (I forgot to get his last name) asking what he would have to do to write for the MOTD. I was

glad to tell him we're always looking for new writers, and that he'd be joining good company. Let's look at those out there!

Somewhere there's another Dale Puckett, another Brian Lantz, and God forbid, even another Bruce Warner (my wife says God couldn't have that strange a sense of humor). My point being that the UG would be the ideal place to get published first. You'll find that you're in the company of writers the likes of Bill Barden, Steve Bjork and Jim Kemp. Steve and Jim are two of the newest members of the UG thanks to our trip to Palo Alto.

Good food and good times

If there's one thing I've been able to learn about community breakfasts, it's that you should expect lousy food. By that standard, we were a failure with our first community breakfast. I even went back for seconds, something I'm not suppose to do.

The next rule is that you must have a bad speaker, so we messed up again. Paul Searby was tremendous!

Thirdly, you have to pick on Esther Puckett. Esther couldn't make the breakfast because someone had picked up the wrong bag at the airport (Esther's clothes were somewhere in San Francisco all weekend while she was in Palo Alto). Remember that when you think you're picking up that all too familiar looking luggage at the baggage pick-up.

So by the rules of community breakfasts, we failed, but we certainly had a great time, and everyone enjoyed it. Check the article in this issue on Paul's talk.

So what's next? Keep reading the MOTD. I've heard rumors about a British OS-9 conference some time this summer. It may be just a rumor, but we all know how rumors are. Every so often there's a little truth to them.

Is there anything else to say

I could go on for days, but I'd lose both my friends, so I'll wrap up for now. "Tune in again next month when Bruce says . . . "Oh no, not another MOTD!"



April, 1986

Vol. 2, No. 3

MOTD is published monthly as the official newsletter of the National OS-9 Users Group, 9743 University Ave., Suite 330, Des Moines, IA 50322

Editor

Bruce Warner
OS-9 Users Group
ATTN: Editor

9743 University Ave., Suite 330
Des Moines, IA 50322

President, National OS-9 Users Group

Brian Lantz

Vice President

Bill Turner

Treasurer

Steve Odneal

Secretary

Dave Gibson

Director-at-Large

Dale Puckett

Software Librarian

Dave Kaleita

Finance Committee

George Dorner — Chairman

Membership Committee

Joe Dubuc — Chairman

Public Relations Committee

Bob Rosen — Chairman

Software Exchange Committee

Carl Kreider — Chairman

Software Reviews Committee

Marcus Johnson — Chairman

Subscriptions to the MOTD are free to members of the OS-9 Users Group. Non-members may subscribe by writing to the chairman of the membership committee and paying an annual fee of \$20. Send to: OS-9 Users Group, ATTN: Membership, 9743 University Ave., Suite 330, Des Moines, IA 50322.

MOTD is produced and distributed by Falsoft, Inc., publishers of THE RAINBOW, the official magazine of the National OS-9 Users Group.

OS-9 is a registered trademark of Microware. There is no connection between Microware and The National OS-9 Users Group, nor is there any connection between Falsoft and The National OS-9 Users Group.

CAN I PLACE AN AD?

It hadn't dawned on me that there are a lot of people out there that want to advertise in the MOTD that aren't already advertising in other publications. So what do they do?

Well, here's the deal! You start out by getting your ad made up (called camera ready copy). When that is completed, you submit your ad with a check made out to

the OS-9 Users Group and send both to:

Editor, the MOTD
c/o Bruce N. Warner
14503 Fullerton Road
Dale City, VA 22193-2034

The price for advertising in the MOTD is based on the size, location and number

of colors (limited to black and red). This chart explains more completely.

Make sure that you've included your camera ready copy and a check for payment made out to the OS-9 Users Group. The deadline is the first of the month preceding the month of intended publication.

SIZE	REGULAR		SPECIAL (back cover)	
	1-Color	2-Colors	1-Color	2-Colors
Full Page	\$400	\$480	\$500	\$600
Half Page	\$200	\$240	\$250	\$300
Quarter Page	\$100	\$120	\$125	\$150
Eight Page	\$ 50	\$ 60	\$ 62.50	\$ 75

MOTD Advertising
c/o Bruce N. Warner
14503 Fullerton Road
Dale City, VA 22193-2034
Phone (703) 670-4962
(Include check payable to the
OS-9 Users Group)

More Incredible!

Now there are **THREE!**



TERMINAL NOT INCLUDED

The QT

The *QT* family of multi-user, multi-tasking computers supports from 4 to 20 users. Currently 9 models are available, ranging in price from \$ 1,595 to \$ 8,795. Models are available with the Motorola 68008, 68000 or the new 32 bit 68020 CPU. CPU speeds range from 8 Mhz to 16.67 Mhz; RAM size from 128K to 2048K and ROM from 2K to 256K. All the *QT*'s have a built in SASI interface and will support any hard drive. All *QT*'s include OS9/68K, the multi-user operating system with Basic, utilities, word processing and spreadsheet programs. The *QT*'s take up less than one cubic foot of space.

The QT series:

The basic *QT* has 128K RAM, 68008 CPU, 8 Mhz and will support 4 users and 2 printer ports. The single floppy version is priced at \$ 1,595 (Retail \$ 2,095) and is field upgradeable to 512K RAM and 20 Meg hard drive. This system sells for \$ 2,995 (Retail \$ 3,595).

The QT Plus series:

This *QT* has 512K RAM, 68000 CPU, 10 Mhz and supports 4 users and 2 printer ports. The single floppy version is priced at \$ 2,095 (Retail \$ 2,695) and is field upgradeable to 1024K, 8 serial ports and hard disk. The *QT Plus* 4 user system with 512K RAM and 20 Meg hard drive is priced at \$ 3,495 (Retail \$ 3,995). The 512K upgrade costs \$ 395 (Retail \$ 495).

OS9 is a trademark of Microware Inc.

QT 20 68020
 QT Plus 68000
 QT 68008



The QT 20 series:

This *QT* has 2048K RAM, 68020 CPU, 12.5 Mhz and supports 4 users and 1 printer port. This system can be expanded to 20 users with 16.67 Mhz. The *QT 20* with a 20 Meg hard drive sells for \$ 7,495 (Retail \$ 8,795).

QT Price List 1986

CONFIG.	DIRECT	RETAIL
<i>QT</i> 1 Drive	\$1,595	\$2,095
<i>QT</i> 2 Drives	\$1,750	\$2,295
<i>QT</i> 20 Meg HD	\$2,995	\$3,595
<i>QT+</i> 1 Drive	\$2,095	\$2,695
<i>QT+</i> 2 Drives	\$2,250	\$2,895
<i>QT+</i> 20 Meg HD	\$3,495	\$3,995
<i>QT 20</i> 20 Meg HD	\$7,495	\$8,795

OS9/68000 SOFTWARE Available Now

Sculptor	\$995 (\$695 for <i>QT</i> owners)
Microware C	\$400
Microware Pascal	\$400 Add 3.50 Shipping

1976 **FHI** FRANK HOGG LABORATORY 1986

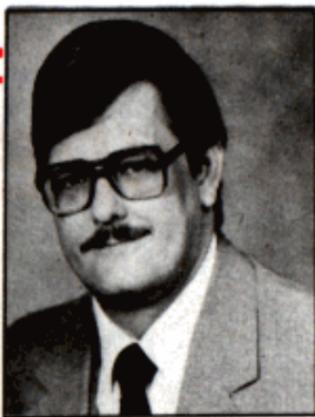
770 James St., Syracuse, NY 13203 Telex 646740

315/474-7856

Continued from page 1

```
00011      ****
00012      * Strip [+/-1] [-c] [+b] [+d]
00013      *   Copies standard input to standard output, removing
00014      *   control characters (except carriage returns and line feeds)
00015      *   as it goes.
00016      * "+1" option causes line feed to be added after each carriage retu
00017      * "-1" option causes line feed characters to be filtered also.
00018      * "-c" option causes carriage return characters to be filtered also
00019      * "+b" option causes backspace carhacters to be processed.
00020      * "+d" option causes stripped characters to be sent to the error ou
00021      * Written by Darryl A. Hock and David L. Kaleita, 02/07/84
00022      * Placed in the public domain, 02/19/85
00023      0000 87CD03D6      mod  STPEND,STPNAM,PRGRM+OBJECT,REENT+1,STRIP,S
00024      000D 53747269      STPNAM fcs  "Strip"
00025      0012 04          fcb   4      Edition number
00026      000D          C$CR   set  $0D      ASCII carriage return
00027      000A          C$LF   set  $0A      ASCII line feed
00028      0008          C$BS   set  $08      ASCII back space
00029      0043          C$ASCC set  'C      carriage return parameter char
00030      004C          C$ASCL set  'L      line feed parameter character
00031      0042          C$ASCB set  'B      backspace parameter character
00032      0044          C$ASCD set  'D      display stripped characters
00033      ****
00034      * Static Storage Offsets
00035      D 0000          org   0
00036      D 0000          BUFSIZ rmb  2      Size of input buffer
00037      D 0002          BUFPTR rmb  2      Output buffer pointer
00038      D 0004          BYTCNT rmb  2      input buffer transfer size
00039      D 0006          ORDER  rmb  1      parameter order temporary
00040      D 0007          MINUSC  rmb  1      1= strip carriage returns
00041      D 0008          MINUSL  rmb  1      1= strip line feeds
00042      D 0009          PLUSL   rmb  1      1= add line feeds
00043      D 000A          PLUSB   rmb  1      1= process backspaces
00044      D 000B          PLUSD   rmb  1      1= display stripped characters
00045      D 000C          FLAG    rmb  1      1= 1st stripped character
00046      D 000D          CHAR    rmb  2      Displayed character string
00047      D 000F          STRPCNT rmb  2      Number of bytes stripped count
00048      D 0011          STACK   rmb  $250-.  Stack space
00049      D 0250          BUFBEG  rmb  4096-.  I/O buffer (May be increased)
00050      D 0000          STRPMEM equ
00051      0013 5573653A      STRPUSE fcc  "Use; Strip [-c] [+/-1] [+b] [+d]"
00052      0034 0A          LF     fcb  C$LF
00053      0035 57696C6C      fcc  "Will strip all control characters except
00054      006C 0A          fcb  C$LF
00055      006D 20207374      fcc  " standard input and send result to stan
00056      00A1 0A          fcb  C$LF
00057      00A2 20202D63      fcc  " -c = Strip carriage returns from input
00058      00CB 0A          fcb  C$LF
00059      00CC 20202D6C      fcc  " -1 = Strip line feeds from input."
00060      00EF 0A          fcb  C$LF
00061      00F0 20202B6C      fcc  " +1 = Add line feeds after each carriag
00062      0121 0A          fcb  C$LF
00063      0122 20202B62      fcc  " +b = Process back space characters."
00064      0147 0A          fcb  C$LF
00065      0148 20202B64      fcc  " +d = Display stripped characters to er
00066      017B 0D          fcb  C$CR
00067      0169          STRPSIZ set  *-STRPUSE
00068      017C 0A          STRIPD fcb  C$LF
00069      017D 53747269      fcc  "Stripped characters:"
00070      0191 0A          fcb  C$LF
00071      0192 24          fcc  "$"
00072      0017          STRPDSZ set  *-STRIPD
00073      0193 0A          TERM  fcb  C$LF
00074      0194 4E756D62      fcc  "Number of bytes stripped: $"
00075      001C          TERMSIZ set  *-TERM
00076      01AF 2C24          SEP  fcc  ", $"
00077      0002          SEPSZ set  *-SEP
00078      ****
00079      * Move stack and setup buffer space
00080      *
00081      01B1 32C90211      STRIP  leas  STACK+$200,U Move stack to safe place
00082      01B5 3440          pshs  U      Bottom of memory
00083      01B7 1F20          tfr   Y,D    ..from top of memory
00084      01B9 A3E1          subd  ,S++   ..equals memory size
00085      01BB 830250          subd  #BUFBEG ..less buffer space
00086      01BE 108E0000      ldy   #0     divide buffer space by 3
00087      01C2 830003      STRIP00 subd  #3     /
00088      01C5 2504          bcs  STRIP01 /
00089      01C7 3121          leay  1,Y    /
00090      01C9 20F7          bra  STRIP00 /
00091      01CB 109F00      STRIP01 sty  BUFSIZ Store buffer size
```

Continued to page 12



UNIX, THE OS-9 LOOKALIKE

by Brian A. Lantz

What does everything that OS-9 does, but does it slower, less dependably, and less efficiently than OS-9? You guessed it. UNIX. If you ever have the opportunity to use UNIX, you will feel very comfortable there. All of the OS-9 goodies are there, they just take longer to execute.

And you will rarely see a floppy-based UNIX system. That's because the UNIX kernel is not as efficiently written as OS-9. It is several times larger. It requires a hard disk to make it's swapping of tasks even slightly useful. But that's okay, UNIX is an evolving system! I'm sure that as they perfect it, it will become more "OS-9 like."

Sure, the last two paragraphs are a bit "tongue-in-cheek." Yeah, they may have stretched the truth a bit! But why do you always hear about UNIX, and never hear about OS-9 in the generic computer magazines?

And some of our own manufacturers don't help the situation. Some of them would like to hide the fact that they run OS-9. Some of our manufacturers want to hide the fact that OS-9 is as flexible as it is; running on systems from a CoCo, to several SS50 manufacturers, and on to VME-bus systems. Systems from a single board, to a complex system of various components.

Instead of hiding this, we should be shouting it. We should be letting the world know that for any application (from industrial control systems to multi-user business systems) OS-9 is THE system. When the Apple Macintosh first came out, Apple sent out several people to promote the Mac. These people were dubbed the "Mac evangelists", because their mission was to win the world over to the Macintosh. We need to be "OS-9 Evangelists", telling the MS-DOS'ers and the UNIX'ers and the CPM'ers that there is a better way.

And while we are at it, let's drop this silly battle within our own ranks that has been continuing for so long that most of us can't remember a time of peaceful co-existence. I'm talking about the old "If you don't own a ***** computer, then it's not any good"! The "war of the hardware" has done more to damage OS-9, the SS50 bus, the Motorola family of processors and the UG than all other combined factors. I am firmly convinced that if we (notice that I include myself in this), if we will unite together for the common good, we will be a stronger power in the marketplace. Let's get the "us versus them" out of our dealings with each other! The OS-9 Users Group is for the promotion of OS-9, not the promotion of any particular vendor's hardware or software.

THE "AT LAST, SOMEONE'S DONE IT RIGHT" DEPARTMENT:

(Oh, boy. With sub-titles like this, I feel like Dvorak!) Someone has finally designed a set of peripherals for the CoCo that allow you to take advantage of the flexibility of OS-9. Check out the ad on the back cover for one of our newest advertisers, CRC Computers and DISTO. For an example, in one slot of a multi-pak you can have a floppy disk controller, a real time clock, a parallel printer port, and an 80 column display. All this and 3 slots left over! Now let's

see . . . If we can get this much into a single slot, how much can we pack into the other 3 slots?

GROWING PAINS:

The single most difficult thing to adjust to is "change." We resist it personally. We avoid it professionally. In most cases, we allow our apprehension about "change" to limit us in our usefulness.

I say this, because the UG is still incurring "growing pains"! The membership of the UG is still growing at a rate of about 100 new members a month. The activities that we are involved in or responsible for are growing daily. The MOTD, as you have been able to see over the last few months, have improved in quality, grown in size, and is trying to give you the information and features that you have been asking for.

With all of this, and the few things that we are not yet able to discuss, the UG officers are having to try to organize themselves better and involve more people. If you have a desire to assist in operating some area of the UG, let the officers know.

I am very pleased that so few pieces of correspondence have "slipped through the cracks". I can count on the fingers of one hand the number of letters of complaint we have received, for one reason or another. But even one unsatisfied member is one too many! Help us to keep this to a minimum in these two ways: (1) Whenever you correspond with us, use the new UG address, not the old P.O. Box. Though this mail gets forwarded to us, it takes it a lot longer to get to us. In a few cases, the delay has been several weeks. (2) Always include your telephone number. Under most cases, a misunderstanding can be straightened out with a quick phone call.

While I am on the subject of "growing pains", let me mention a problem that we had with the last issue of MOTD that you may not have been aware of. The problem, one that Tim Grovac would have loved to have, was that we had too much copy for the last issue, and some of the articles had to be left until this issue. These kind of problems are good to have, since they say that there are more people contributing more articles! If these kind of problems persist, we may need to increase the size of the MOTD. In the meanwhile, our thanks to Marc Johnson, Fred Scoville, and Bert Schneider for their understanding with this situation.

ABOUT COMPUSERVE:

And for your information, if you haven't been on the OS-9 SIG of Compuserve in the last few weeks, there have been some changes. The SIG has a new SYSOP, Wayne Day. Wayne is assisted by two capable assistants, Jonathan Clutz and Pete Lyall. While it's still the "Same 'ole SIG", Wayne promises bigger and better things for the OS-9 spot of Compuserve.

THE "YOU ARE WHAT YOU WEAR" DEPARTMENT:

The OS-9 UG has the last of the OS-9 tee-shirts that Microware had made for last year's Des Moines Seminar. There

are two types; the clean one, and the "provocative" one. The clean one says "To be good, you have to be different." The "provocative" one says, "Our software keeps your hardware up", and features a female on a street-corner (Nuff said?). These are both great shirts, and a great way for us to advertise OS-9 to the multitudes. If you wish to get one or both of these shirts, send \$7.50 to the UG address, marked "ATTN: MISC." Please indicate which shirt and the size (small, medium, etc.). There are only a few of these left, so if you are interested, act fast! Whatever shirts we have left, we will take to the RainbowFest in Chicago.

SPEAKING OF RAINBOWFEST:

For information on the Chicago RainbowFest, see the article elsewhere in this MOTD. We will be sponsoring another OS-9 Buffet Breakfast on Sunday morning. We are pleased to have Bill Moore, from Microware as our Keynote speaker for the breakfast. Bill's boss, Ken Kaplan, was planning to speak to us at the Chicago breakfast, but Microware has been so busy lately, that he asked if we would let him take a raincheck. With this development, I took no time to ask Bill to speak to us. Bill is in charge of several new areas at Microware, including the End-User Seminars. Bill is a charter member of the OS-9 Users Group. If you live in the Chicago area, you should consider the RainbowFest May 23-25.

At the California RainbowFest, the UG started a tradition at the OS-9 Buffet Breakfasts of giving awards, OS-9 UG Grin Awards. I am proud to say that the first recipient of this award was Steve Bjork, who got the award for the most improved attitude. Steve was dragged (kicking and screaming) into the world of OS-9 by Tandy. Steve had made up his mind that he was not going to enjoy his association with OS-9, at all. But now, several months later, Steve has learned to appreciate OS-9 and love it as we have.

Another Grin Award winners worth noting is Fran McGehee, the Marketing Information Representative in charge of the Color Computer at Tandy Corporation. Fran's Award was entitled "The Twilight Zone Award." This was not a personal slur. Fran received this award for entering the wacky world of the CoCo, and the laid-back community of OS-9 and living to tell about it. I'm not sure what Fran has to think about all of us wierd computer users, after the last RainbowFest.

Also receiving awards, for various reasons, were Wayne Day, Lonnie Falk, John Roach-Ed Juge-Mark Segal-Tandy et al, and Jack Tremell. It may just come to be that the "big names" of the OS-9 world won't dare miss the Buffet Breakfasts of RainbowFest, for fear that they will be the unknowing recipient of the UG's Grin Award.

HOT OFF THE PRESS:

The latest and most exciting news about OS-9 was recently found in the Des Moines Register. The Des Moines local paper reported an interesting story about Sony, Phillips, Microware, and the new standard for interactive CD ROM players. See the article on the CD ROM Standard for the brief highlights. By the next issue of MOTD, we should have a lot more details.

RUMORS:

And to close, let me leave you with a rumor that I heard today. Now understand, this is a rumor! I do not (personally) know if this is true, though I have reason to believe it to be so (half because of the content, half because of the source).

If seems that a relative unknown to the computer industry, a young redheaded man by the name of Bill Gates, was talking to one of the clients of his struggling company, Microsoft. When talking to this client, it is said that the conversation mentioned that OS-9 had become the De-facto Standard operating system for the Motorola 68000 processor family. Mr. Gates assured his client that since this is so, his company (Microsoft) would be doing their best to start porting their major software products to 68000 OS-9.

Steve Bjork, step aside. You no longer have the most improved attitude about OS-9. I think that Bill Gates should now be given this award. Welcome Microsoft, into the world of a TRUE operating system!

CD ROM Standard

The March 4 issue of the Des Moines Register carried an interesting story. It seems that a joint announcement was made locally by Sony, Phillips and Microware, with a national press announcement to follow in a few weeks. Sony and Phillips have been working on a Standard for the newly developing CD ROM industry. These two super-companies called Microware and the three companies have been working together since 1982.

The result is a standard for CD (Compact Disk) Interactive ROM Players. The machines will contain Motorola 68000 processors, and will be run by software containing 68000 OS-9!

The CD ROM Technology is an exciting new frontier, with as many angles as your imagination can think of! Talking encyclopedias, and dictionaries are the most commonly thought of, but why not games, or educational tools, or . . .

These CD ROM players are said to be

as easy to use as a VCR, with the capability to hold the equivalent of over 150,000 full pages of text.

This standard, developed together on three different continents, should start materializing in products by as early as 1987.

What does this mean to OS-9? It's too early to tell how much OS-9 will become a part of other commercial products. It's also too early to tell how important this announcement really is. Many industry watchers are saying that this is as important a decision as was the licensing of MS-DOS by IBM for their first personal computer. Some are saying that this is only the beginning of many important deals involving OS-9. It's too early to tell how much of this is true. Maybe these are understatements!

Look in the next issue of MOTD! We will try to have more details on this exciting and important development in the life of your favorite operating system!

THIS COULD BE A PRETTY PICTURE

Micro Illustrator review by Fred Scoville

Micro Illustrator — A fun program for all ages! What is it? A graphics editor written in 1984 and licensed by TANDY for distribution using OS-9 as its operating system.

Three weeks ago the Northern Virginia Color Computer Club purchased a DMP-220 ink jet printer during one of Radio Shack's sales. Knowing this review was planned an OS-9 screen print program was ordered from the local Radio Shack store but here it is deadline time and no program. The review will not have any pictures but so be it. Maybe a supplement next issue.

The package is a single disk and a manual attractively packaged as always by Tandy. The first thing I did after backing up the disk and scanning the manual was to look for example pictures. Alas no examples. The following lists the directories supplied:

TANDY" created on: 84/12/26
Capacity: 630 sectors (1-sector clusters)

444 Free sectors, largest block 441 sectors

The program is booted as is OS-9 and then at the OS-9 prompt the command

illustrate is issued and Micro Illustrator appears. The documentation supplies a boot program to be typed in for any users who do not have one and do not have DOS1.1 or better. No problem for experienced OS-9 users but my telephone calls and mail show it to be confusing for the novice.

The main menu has fifteen commands which are accessed by using the joystick

to select the icon, pressing the fire button and then the space bar to go to the drawing screen.

Sounds simple but it takes awhile to get used to. Also the disk icon and clear icon immediately go to a submenu when the fire button is pressed. The mirror and zoom function are used in conjunction with the other commands. Three functions edge, line (as opposed to lines) and

text do not have an icon. Sounds confusing. It is, but fortunately there is a better way. All commands have a keyboard implementer which invokes the command with a single keystroke without leaving the drawing screen. This is the best feature of the editor and is enjoyable when compared to other graphic editors even the cadillac of them all—COCOMAX. Viewing the graphics screen while switching palettes with a single key makes this program a great doodler.

In addition to the fifteen icon functions each palette gives a choice of "brush" shape and pattern. With the functions of draw, lines, copy, spray, point, fill, frame, box, circle, zoom, rays, edge, line, and text one can create to his or her delight.

As with most graphic editors unless you have good artistic talent or the patience to trace from a screen overlay I don't think you will be able to emulate VanGough but the program is fun. It doesn't have all the functions of better known editors but it only costs \$30. The menu is clumsy but the single stroke keyboard implementation takes care of this.

There does appear to be one major bug in the program. Pressing the break key locks the computer and all is lost.

d-ewrewr	A 0	120	84/12/26	08:41	CMDS
----wr	23 0	3607	84/12/26	08:53	OS9Boot
---r-wr	5B0	E	84/12/26	08:54	STARTUP
---r-wr	5F0	6CB	00/00/00	00:33	SICONS
-s-wr-wr	67 0	1800	00/00/00	00:09	TITLE
d--wr-wr	A20	40	00/00/00	00:00	PICTURES

--e-rewr	13 0	298	83/02/18	16:29	DIR
--e-rewr	17 0	11D	83/01/14	09:12	RENAME
--e-rewr	1A0	A5	83/02/18	16:29	DEL
--e-rewr	1C0	118	83/01/14	09:12	SETIME
--e-rewr	1F0	2D1	83/04/14	14:24	FREE
--e-rewr	5D0	29	84/10/01	11:40	CLS
--e-rewr	80 0	20E8	00/00/00	00:08	ILLUSTRAT

Accounts Payable/Receivable Systems

a review

by Marcus W. Johnson

What's his opinion worth?
a note from the editor

The author is a software engineer currently working for a company in the defense industry, where he is alternately frustrated by the software development system (a FLEX system (!)) and the company's mainframe (a VAX running VMS). He also does consulting work, helping hapless pawns of Big Blue run their expensive software. He achieves some small measure of satisfaction (and sanity) running a CoCo, with OS-9 of course, and the new 80-track TEAC he got for Christmas (thanks to Brian Lantz for expediting it). He aspires to a 68K system in the near future (towards that end, he accepts contributions in the form of cash or checks).

He is cruel to poorly-written software and once made a CP/M system cry; he's still waiting for the right editor to come along (currently tolerating DynaStar).

Accounts Payable and Accounts Receivable are business software products from Computerware. The copies reviewed were run on a CoCo, using Level 1 OS-9.

The programs consist of applications written to run on Computerware's business BASIC, Random Basic. Taking

advantage of the speed afforded by the random file access capabilities of Random Basic, the programs run much faster than I expected. There are noticeable delays in execution as files are loaded from disk, but in applications such as these, running in a 64K environment, this is unavoidable. These delays typically occur when new menus (and associated programs) are loaded.

The programs perform their basic accounting functions well. While the documentation only provided a rudimentary explanation of how each program worked, I was able, with a rather limited knowledge of accounting, to follow the screen prompts well enough to run each system without incident the first time up. If I were to grade this, I would give the documentation a C-, but the screens get an A.

This is not to say that there were things I wouldn't like to see cleaned up. In my consulting work, I often hear complaints about how older business software (and you'd be amazed — or would you? — at how often software for Big Blue suffers in this respect) can not or will not handle the new 9-digit ZIP codes, or can't handle foreign codes (like Canada, for instance). Alas, these packages fall down in that respect. Under Accounts Payable, I experienced a disk problem (the problem lay in my drive, but the reason is not really relevant); after realizing that

something was terribly wrong, I wound up in Random Basic. Nothing I could do would restore me to my application, and I finally was forced to hit the reset and re-boot. This is not state-of-the-art error handling. Finally, there were little annoyances, such as the fact that on yes-no questions, the program would insist on only upper-case sometimes, and either case at other times. I could go on and you

could say that I'm nit-picking; maybe you're right. But I would have a lot easier time selling it to a business prospect with a little clean-up.

In any case, if you're running a small to medium sized business, don't mind a few idiosyncracies, and need a good accounting system, this is a nice start. At a mere \$125 each, you're getting your money's worth.

Synopsis

Product(s):	Accounts Payable System (Computerware) Accounts Receivable System (Computerware)
Cost:	\$125.00 each
System:	Radio Shack Color Computer; 2 disk drives required; printer required
Documentation:	Weak (*)
Ease of use:	Good (***)
Error handling:	Fair (**)
Performance:	Superior (****)
Overall:	Good (***)

Rating system:	
Poor ()	The family dog might do better
Weak (*)	It's adequate — just barely
Fair (**)	Tolerable; we've seen worse
Good (***)	Does a nice job, but we've seen better
Superior (****)	Flawless
Excellent (*****)	Redefines state-of-the-art

TURNER TALKS BACK!!!!

by Bill Turner

Rain, rain and more Rain. If you were there, you know what I mean. If you weren't, well, maybe, you saw it on the National News. Where? Oh, I'm sorry, I was talking about the February RainbowFest in Palo Alto. Now, I haven't lived in the San Francisco area for at least 18 years (used to live in San Rafael/Novato), but I'm sure that the folks there didn't really order that much rain!

Now if I could really spin a good yarn, I'd have you in stitches with one of those good news / bad news disaster stories — mine kinda goes like this: It was raining when we landed in San Jose. You get to walk to the terminal (down metal stairs and across the apron). We got all of our luggage (7 boxes) and they were the first 7 boxes off the luggage conveyor. Carried the boxes outside to get a cab. The cab can't get closer than 3 lanes away from terminal. Still Raining. The cabbie whips out of the parking lot and onto highway 101. Windows are fogged, including the windshield. Driver learned how to drive watching late night movies of Elliott Ness in New York City (or so he thought). Roll down all windows so cabbie can see. Still raining. Driver finds Palo Alto, but can't find hotel, etc.

But that's OK, 'cause I enjoyed my hour and a half, \$25, 30 minute tour of Palo Alto. I especially liked watching the faces of: The guys walking down the street . . . or the driver of the next car at the stop sign . . . or the passenger in the back of a car at a Gas Station . . . when any of them were asked "Where's the hotel . . . ?" (You see the driver spoke "excellent" English, and . . .) If you ever visit San Jose and/or Palo Alto, I would recommend this tour — It's a real trip! (And I thought the plots that Goldy Hawn and Monty Python used in the movies were fantasies . . . Silly me!)

Actually the show was very interesting and informative — I especially liked the "going out of business garage sale" that was being held by a major vender. What was even better, was their total inability to accept cash for the first hour or so on Saturday morning. They did take plastic though, but why they found it necessary to call in credit verifications on a \$12 purchase made with an American Express card is beyond me. What-the-heck, I was dumb enough to wait twenty minutes for them to run to the hotel lobby to place the call.

On a better note, I did get to meet and talk with quite a few of you, and I talked to quite a few prospective members. Some came by wondering what OS-9 was and why we were at a Color Computer show with a QT. A number of folks stopped by the OS-9 booth just to say hi, and a few came by looking for answers to some problems that they thought were uniquely theirs (the same unique problem 47 times).

For example, a couple of guys stopped by with a BASIC-09 problem on Friday night. Seems that they had been trying to get around a particular error message for some time and were hoping that someone could help. It just happens that I had just solved the same problem a couple of weeks prior to the show, and so I talked with them for a while. All of the sudden the light went on and they took off to return to Northern California to try out the solution.

What really surprised me was when they came back the following afternoon to say "thanks!" I like the way they said it though — with a fifty dollar check and two filled out membership applications. I talked to junior high students and I talked to grandfathers, and several times I found that I was running a mini-seminar in the middle of the aisles with 20 to 30 people around me. What I'm finding out is that there are a lot of people out there with a very large appetite for knowledge about OS-9 and its languages.

Following the show, I received a very nice letter from Vicki Daubner of Redding, CA, thanking me for assistance provided at the show: ". . . I really appreciated the fact that everybody seemed happy to answer questions. That was a whole new experience for me. I'm used to getting silence or evasions. By the time I got back to Redding, my head was reeling and I'm still trying to sort everything out!"

Unfortunately, I see and hear from many people like Vicki — they want answers and what they get frequently is pure garbage. Vicki, now you know why the UG exists and maybe in the future someone will ask you a question, and you'll be able to help!

As a reminder for those of you in the Chicago area, The UG will also have a booth set up at the Chicago Rainbowfest in May. If my memory serves me correctly, it will be at the Hyatt Regency — Woodland. Hope to see you there.

Which brings me back to my favorite subject — UG's and why they exist. It's simple. Help. Assistance. Support. Knowledge. The newcomers learn by asking the old-timers. The old-timers learn more by trying to help the newcomers. As Steve Bjork said, "some of us remember what it took to get here . . ." Right on Steve!

While I think about it — Did you know that many of the programs written to run under OS-9 will run without modification on any of the versions of OS-9 that currently exist? You didn't know that OS-9 does exist on many different kinds of hardware. And that both proprietary single board machines and bus oriented systems exist? Lets see, you can run OS-9 on 6809's, 68000's, 68008, 68010 and 68020 systems. You can run it in s-100 bus systems, in Apple IIs, and even in the IBM-PC family. Let's also not forget that the 6800/6809 SS-50 bus system is also well supported.

You can spend as little as \$300 to \$500, or you can spend 15 to 20 thousand dollars for a system that will run OS-9. And many programs will run on any of them!

The only rules that you have to follow are:

- 1) Use only standard OS-9 calls,
- 2) Minimize your use of assembly language,
- 3) Don't use machine features that are unique to a single manufacturer, if you want to market your program to a larger user base. For instance, The Radio Shack Color Computer supports a nice graphics interface, but at this time it is unique to the CoCo. Some day MICROWARE will be able to retrofit their higher level graphics interface to the smaller machines, but right now it's only available for the HITACHI graphics controller.

A good safe way to write a program that can cross machine boundaries is to write it in one of the higher level languages, such as BASIC-09, PASCAL, FORTRAN, or C. Most, if not all, programs written in these languages will run without modification as long as the program will fit into the address space available, and providing it does not internally call OS-9 commands via the SHELL. (The command syntax for some commands is different for the 6809 and 68K versions of OS-9.)

I believe that the address space availability on a 6809 Level-I system is around 42K. On a 6809 Level-II system it is usually around 62K. I'm not aware of any address space limitations (beyond the actual size of your real memory) in a 68000 system.

The project for this month is to compile a list of all error codes issued by OS-9 programs, especially those issued by OS-9 and the OS-9 language systems. What I am trying to do, is to create a reference document, which would be available to all UG members, that a programmer could use for assistance on what to look for when the operating system issues an error message. For example: If you receive the "WHAT?" prompt after typing in a command name, you should first check to see if that program exists by that name in your current execution directory. If it does, then issue an "IDENT" command to verify that the module you want is the first module in that file. If it is, then also check to see if the "Public Execute" (PE) and/or "Public Read" (PR) bits have been set with the ATTR command. (Oh, by the way this all assumes that the module is not currently in memory — use the MDIR command to verify this.)

A second example: I have found on my GIMIX-III, Level-II system that when BASIC-09 issues an "Error 43" (Unknown Procedure) that it really means

one of the following:

1) You said "RUN XXX" and no module XXX can be found in either main memory, or in the current execution directory.

2) module "XXX" can not be mapped into the memory left in your 64K address space.

3) The program can be mapped, but there is not sufficient memory for the required data areas. (You would think that you would get "insufficient memory" but this doesn't always occur.

4) Basic-09 thinks that there isn't enough memory available.

BASIC-09 adds all of the data and procedure space requirements before it invokes the program. I converted a very large program into a main-line program and a series of callable modules, each having their own temporary variables. The program design is such that only 1 out of the ten modules can actually be active at one time - but BASIC09 assumed that they all needed at least ONE complete data area each, and refused to run my program. The fix was to create a new series of temporaries in the main-line program and to pass the address of them as a parameter each time one of the called programs was run. This slowed down the execution of the program to the extent that the project was scuttled.

5) Another "not enough memory" meaning for an error 43 is as follows: A program which runs in 64K might fail if called from another program, even when it is invoked through a shell, if parameters are being passed. (Apparently the passed parameter is being mapped into a 2K page, thus reducing the amount of main memory available to the called program to 62K.)

Anyway, with these two examples, you should get the idea of what I want to compile. Please send any "hints and techniques" to our Des Moines address and mark the letter "ATTN: Vice."

TEN MOST-ASKED QUESTIONS about DYNACALC®

THE ELECTRONIC SPREAD-SHEET FOR OS-9 SYSTEMS

- 1. What is an electronic spread-sheet, anyway?**
Business people use spread-sheets to organize columns and rows of figures. DYNACALC simulates the operation of a spread-sheet without the mess of paper and pencil. Of course, corrections and changes are a snap. Changing any entered value causes the whole spread-sheet to be recalculated based on the new constants. This means that you can play "WHAT IF?" to your heart's content.
- 2. Is DYNACALC just for accountants, then?**
Not at all. DYNACALC can be used for just about any type of job. Not only numbers, but alphanumeric messages can be handled. Engineers and other technical users will love DYNACALC's fifteen digit math and built-in scientific functions. With 6809 DYNACALC, you can build worksheets as large as 256 columns or 256 rows (18278 columns or 9999 rows in 68000 versions). There's even a built-in sort command, so you could use DYNACALC to manage small data bases — each row of the worksheet is one record.
- 3. What will DYNACALC do for ME?**
That's a good question. Basically the answer is that DYNACALC will let your computer do just about anything you can imagine. Ask your friends who have Lotus 1-2-3, VisiCalc, or similar programs, just how useful an electronic spread-sheet program can be for all types of household, business, engineering, and scientific applications.
- 4. Do I have to learn computer programming?**
NO! DYNACALC is designed to be used by non-programmers, but even a Ph.D. in Computer Science can understand it. Built-in HELP messages are provided for quick reference to operating instructions.
- 5. Do I have to modify my system to use DYNACALC?**
Nope. DYNACALC uses any standard OS-9 configuration, so you don't have to spend money on another CPU board or waste time learning another operating system.
- 6. Will DYNACALC read my existing data files?**
You bet! DYNACALC has a beautifully simple method of reading and writing data files, so you can communicate both ways with other programs on your system, such as the Stylo-Graph word processor, SortMerge, data base systems, or other programs written in Basic09, C, PASCAL, FORTRAN, and so on.
- 7. How fast is DYNACALC?**
Very. DYNACALC is memory resident, so there is no disk IO to slow things down. The whole data array (worksheet) is in memory, so access to any point is instantaneous. DYNACALC is 100% machine code for blistering speed.
- 8. Is there a version of DYNACALC for MY system?**
There's a version of DYNACALC for EVERY OS-9 system. Unless you have a CoCo, you need a CRT terminal with at least 80 characters per line, and direct cursor addressing. You can mix different brands of terminal on the same system. The CoCo OS-9 version is compatible with 80-column hardware cards, or will work with the standard 32x16 screen.
- 9. How much does DYNACALC cost?**
Radio Shack sells the CoCo OS-9 version for \$99.95. The general 6809 OS-9 version is priced at \$250, \$595 for the OS-9/68K version. Foreign orders add \$10 per copy for postage. We encourage dealers to handle DYNACALC since it's a product that sells instantly upon demonstration. Call or write on your company letterhead for more information.
- 10. Where do I order DYNACALC?**
If you have a CoCo, order # 26-3275 at your local Radio Shack store. Otherwise, see your local DYNACALC dealer, or order directly from CSC at the address below. We accept telephone orders from 10 am to 6 pm, Monday through Friday. Call us at 314-576-5020. Your VISA or MasterCard is welcome. All orders are shipped on 5" diskette unless you specify otherwise. Please tell us if you need Microware (standard) or Mizar format.

Computer Systems Center
42 Four Seasons Center #122
Chesterfield, MO 63017 USA
(314) 576-5020



DYNACALC Reg. U.S. Pat. Off.
OS-9 and Basic09 are trademarks of Microware & Motorola.
1-2-3 is a trademark of Lotus Development Corp.
VisiCalc is a trademark of VisiCorp.
Stylo-Graph is a trademark of Stylo Software, Inc.
Radio Shack is a trademark of Tandy Corp.

DYNACALC OS-9

OS-9 U.G. SOFTWARE COMMENTARY

by Bert Schneider

This month I'll be discussing five of the OS-9 User Group Disks. I have decided to change the stepping rate of this article to a faster one so those of you who are anxious to read about disk numbers in the upper 30's and 40's won't have to wait until OS-9 Level XXIV comes about. The topics of discussion this month are; Disk #3, Word Processing Utilities; Disk #4, Programming Utilities; Disk #5, File Processing Utilities; Disk #6, Adventure Source; and Disk #7, "Adventure Program."

Disk #3 Word Processing Utilities

There are three categories of programs in this issue; programming aids, text file processing and text file output routines. For a change of pace, I will use a different format from my previous "text file output."

NAME: split
TYPE: text file output
LANGUAGE: C **SIZE:** \$2089
DESCRIPTION: This neat utility provides the user with a side by side reference of two text files. Only the first 40 columns of each text file are displayed onto the standard output. The program makes no comparison of the files, that is left up to the user (or one of the other utilities on this disk). I find this particular utility to be the most useful for quick and dirty comparisons.
USE: split file1 file2 <cr>
AUTHOR: Eric Williams

NAME: sqsh
TYPE: text file processing
LANGUAGE: 6809 source
SIZE: \$2B6
DESCRIPTION: This program replaces each carriage return with a slash ("/"). The line size can be anywhere from 1 to 32767 as specified in the command line (the default is 80). Errors may result if the user gives an incorrect line size.
USE: sqsh <input pathname [NNNNN] <cr>
 where NNNNN is line length (default to 80)
AUTHOR: Peter Dibble

NAME: tab
TYPE: text file processing
LANGUAGE: C
SIZE: \$579
DESCRIPTION: This routine converts spaces in a text file into a combination of tabs and spaces. The program assumes a tab size of 8 spaces. If no names are given, then tab looks for text from the standard input. It should be noted that tab cannot differentiate between spaces in normal use and those in literal strings. This program is great for file compaction (especially "C" routines).
USE: tab file output—file <cr>
 or
 tab <cr>
AUTHOR: Eric Williams

NAME: tc
TYPE: programming aid
LANGUAGE: C
SIZE: \$F8F
DESCRIPTION: This utility compares two text files and outputs to the standard output the line numbers that differ between

the two files. In addition to the line number, the program lists what each file has on that particular line. For example,

```
.in 10
10=This is a test.
10=This was a test.

13=Another neat test.
13=Maybe another test.
.in
```

is the result of two files that have different text on lines 10 and 13. **USE:** tc file1 file2 <cr>
AUTHOR: Carl Kreider

NAME: texcom
TYPE: programming aid
LANGUAGE: BASIC09
SIZE: \$162
DESCRIPTION: Texcom compares two text files, one line at a time. It performs the same function as tc.
USE: texcom <cr>
(program is self prompting)
AUTHOR: Carl Kreider

NAME: translit
TYPE: text file processing
LANGUAGE: BASIC09
SIZE: \$E3F
DESCRIPTION: This routine transliterates from one alphanumeric to another. According to The American Heritage Dictionary of the English Language, transliterate means "To represent (letters or words) in the corresponding characters of another alphabet." There are many uses for a program like this. First, you could make a cipher and code all of your correspondence. But more realistically, you can use translit to change certain keystrokes to suit your own particular tastes. For example, in a text editor, you may prefer "l" instead of "f" for "locate" instead of "find." This method of creating your own aliases makes a home computer even more personal by tailoring the input to each individual user.
USE: translit >/p <cr>
 or
 translit >/file <cr>

This method of invoking translit will prompt you for input. You are expected to provide translit with a set of alphanumerics that you want changed. For example:

? abc/def (note, you must include a slash between sets)

will change a to d, b to e, and c to f.
AUTHOR: Laura Bialon

NAME: untab
TYPE: text file processing
LANGUAGE: C
SIZE: \$3FC
DESCRIPTION: This program is used in conjunction with tab. It expands each tab into 8 spaces.
USE: untab file output_file <cr>
 or
 untab <cr>
AUTHOR: Eric Williams

NAME: words
TYPE: text file processing
LANGUAGE: 6809 source
SIZE: \$F9

DESCRIPTION: This routine outputs each word in a text file to a separate line. I renamed my routine to just plain "word" so as not to confuse this program with "words" found in the dictionary disk, volume #1. It will take the following sentence "This is a test", and create this:

```
This
is
a
test
```

USE: file ! words <cr>
AUTHOR: Peter Dibble

Disk #4 Programming Utilities

Disk #4 has three areas of programming utilities. File maintenance, programming aid, and system utilities provide the user with capabilities to manipulate files, program and test more efficiently, and to use the operating system to its fullest extent possible.

NAME: pwd
TYPE: system utility
LANGUAGE: 6809 source
SIZE: \$5F1
DESCRIPTION: The major difference between this utility and the OS-9 pwd is the ability of the user to specify the number of levels of directories printed out. By just giving the command only the current directory (working, execution denoted by using an x) is listed rather than the entire path. The path to the directory can be listed by using the * option. You now only require one utility instead of two (pwd and pxd) programs which are both 304 bytes long vs. 1521 bytes. Note: I was unable to assemble this program without warnings and it would not run on my system (Color Computer). I have volume 4.01 and perhaps there is a limitation in this version to run on a color computer.

USE: pwd <cr>
 DIRECTORY—NAME

```
pwd x
CMDS
```

```
pwd x *
/d0/CMDS
```

```
pwd 2
ASM.6809/DOCUMENTATION
```

```
pwd *
/D1/SRC/ASM.6809/DOCUMENTATION
```

AUTHOR: Peter Dibble

NAME: bincom
TYPE: programming aid
LANGUAGE: BASIC09
SIZE: \$15F
DESCRIPTION: This routine performs a byte for byte comparison of two specified files. Bincom reports differences and the offset into the file to the standard output.
USE: bincom <cr> (it is self prompting)
AUTHOR: Carl R. Kreider

NAME: modbuild
TYPE: programming aid
LANGUAGE: BASIC09
SIZE: \$259

DESCRIPTION: This program builds a multimodule file by selecting from existing multimodule files. It provides a good routine for making a new OS-9 boot file. The program asks for an output file name and then repeats queries for input filenames until "q" is entered. After a filename is entered, the header and size are printed and modbuild prompts the user with "Y" or "N." Thus the user can select which modules will be included.
USE: modbuild <cr>
AUTHOR: Hal Snyder

NAME: graft
TYPE: file maintenance
LANGUAGE: C

SIZE: \$AD3
DESCRIPTION: This utility enables the user to copy a sub-tree of the directory structure. The copy is performed by the program rather than creating a shell input file. This saves time and memory. It contains a 32K buffer and all file attributes, names, last access dates, etc. remain the same. Another neat feature of this routine is that it is self-documented. If you just type in graft it will explain the use of the program. All programs should include an option like this.
USE: graft source_directory target_directory
AUTHOR: Eric Williams

NAME: latest
TYPE: system utility
LANGUAGE: C
SIZE: \$1865
DESCRIPTION: Latest prints a path name of files whose last modified date and time is later than the one given in the command line. The output is directed to the standard output. If the date is omitted, the current date is used. If no time is given, then midnight of the given date is used. This is a very powerful program especially for configuration management type purposes.
USE: latest directory_name [date] [time]

date format: [mm/dd[yy]]
time format: [hh:mm]

AUTHOR: Eric Williams

NAME: module
TYPE: system utility
LANGUAGE: C
SIZE: \$1E86
DESCRIPTION: Module provides the capability to produce a module directory of a particular executable file. For example, a compiled BASIC09 program may have several procedures packed in a file. All you see with the dir command is the filename (even with the e option). This program can also act as a filter.
USE: module /pathname A list of names and offsets of each module appearing in the standard input are printed out. The module CRC is checked and execution will end if an error is found. Example:

```
module /d0/cmds/checkbook
0000 checkbook
033A getchecks
0458 getdeposits
```

AUTHOR: Eric Williams

NAME: mv
TYPE: file maintenance
LANGUAGE: C
SIZE: \$A35
DESCRIPTION: This little gem moves files from one directory to another. It even copies large files quickly by manipulating the pointers. Both the file and target directory must be on the same physical device however. This is also another self-documented program. This feature comes in handy if you don't have a hard disk with an entire HELP directory on it.
USE: mv file directory
AUTHOR: Eric Williams

NAME: dcopy
TYPE: file maintenance
LANGUAGE: BASIC09
SIZE: \$B63
DESCRIPTION: Directory copy utility.
USE: dcopy (self prompting)
AUTHOR: Brian Capouch

Disk #5 File Processing Utilities

NAME: stripz
TYPE: text file processing
LANGUAGE: BASIC09
SIZE: \$CE
DESCRIPTION: This routine copies all lables in a disassembly listing with a prefix of "z", such as zloop1. The output is put into a separate file for creation of a substitution file with an editor.
USE: stripz <cr> (self prompting) AUTHOR: Carl Kreider

NAME: equfix
TYPE: text file processing
LANGUAGE: BASIC09
SIZE: \$1D4
DESCRIPTION: Equfix strips comments, blank lines, and pseudo opts from assembler files. It does a very good job and is very handy for making room for additional memory required to assemble a program.
USE: equfix <cr> (prompts user for input and output files)
AUTHOR: Carl Kreider

NAME: pad
TYPE: text file processing
LANGUAGE: BASIC09
SIZE: \$162
DESCRIPTION: This program improves the readability of disassembled listings by inserting blank lines after control transfer statements, etc. Self prompting.
USE: pad <cr>
AUTHOR: Carl Kreider

NAME: sort
TYPE: file processing filter
LANGUAGE: C
SIZE: \$3E1
DESCRIPTION: This is a general purpose sort routine. It sorts lines of text from the standard input and produces a sorted version out to the standard output. The file must fit in memory. This is a very quick all purpose sort program that can be implemented into just about any program.
USE: sort <file> >sorted_file
AUTHOR: Eric Williams

NAME: hx
TYPE: file processing filter
LANGUAGE: 6809 source
SIZE: \$430
DESCRIPTION: This routine converts the standard input to a readable Hex dump format output along with the ASCII equivalent. The x option will suppress the ASCII output and the line size can be changed by giving a number from 1 to 32767. Give it a line length of 0 and hx will act as a pipe for modifying the output of another program.
USE: hx /d0/cmds/program
AUTHOR: Peter Dibble

NAME: stripnum
TYPE: text file processing
LANGUAGE: BASIC09
SIZE: \$345
DESCRIPTION: Another self prompting program that strips the first "N" characters from the beginning of each line of text. The input file is not changed. Rather, a new output file is created to keep the original intact.
USE: stripnum <cr>
AUTHOR: Dave Kaleita

NAME: striprem
TYPE: text file processing
LANGUAGE: BASIC09
SIZE: \$550
DESCRIPTION: Dave has come up with another great program that removes the remarks from a BASIC09 source file. It too is self prompting and removes all REM statements from the input file and creates a new output file without them. It only works with REM type remarks. It does not recognize the format of (* *) type remark statements.
USE: striprem <cr>
AUTHOR: Dave Kaleita

NAME: flexbin
TYPE: binary file processing
LANGUAGE: BASIC09
SIZE: \$538
DESCRIPTION: Flexbin converts Flex-format binary files into full image (full size) binary files. It creates a new output file and does not modify the input file.
USE: flexbin (self prompting)
AUTHOR: Dave Kaleita

NAME: newstrip
TYPE: file processing filter
LANGUAGE: BASIC09
SIZE: \$13F
DESCRIPTION: Newstrip removes all control characters except carriage returns from the standard input path.
USE: newstrip or newstrip file
AUTHOR: Dale Puckett

NAME: extract
TYPE: programming aid
LANGUAGE: C
SIZE: \$DB0
DESCRIPTION: This routine will filter a single C function definition including heading comments from a C source file and will output to the standard output.
USE: extract file function name
AUTHOR: unknown

NAME: patch
TYPE: programming aid
LANGUAGE: C
SIZE: \$2708
DESCRIPTION: This program will change selected bytes in a file. Patch will ask for an address from which you can step through the next byte, go backwards, go to another address location, or modify the byte.
USE: patch program
AUTHOR: Eric Williams

NAME: intruder
TYPE: programming aid
LANGUAGE: C
SIZE:
DESCRIPTION: This program is similar to Dump except that selected sectors are displayed as well as contents of directories.
USE: intruder file
AUTHOR: Eric Williams

Disk #6 Adventure Source

This disk contains all the the source code required to compile the adventure "Colossal Cave" into object form. It also contains all of the data text files used by the program. This set of code is so large the Users Group put it on a double sided disk. I had to use SDisk to read the disk because it was stored in standard OS-9 format. I have since copied it over to the Tandy OS-9 format (18

sectors per track double density). When this volume first came out, many people had problems reading the disk. However, the efforts of people like Dave Kaleita and Frank Hogg Laboratory helped to turn things around. I have had barely enough time to play the game let alone look through the code. It has been adapted to Microware C by J. Siefer. The original program, "Colossal Cave", was written by Willie Crowther and most of the current features were added by Don Woods. I can remember when I was going to college we use to play this game on a Burroughs mainframe computer. Oddly enough, the code was written in Fortran which is not suppose to be a string oriented language! Anyway, the source code is fantastic whether you are interested in adventure programming, learning about C, or you just want to find out how to win. You can compile this disk (although I have not tried on my Color

Computer) or you can just procure Disk #7, Adventure Program.

Disk #7 Adventure Program

I highly recommend this program to anyone who has or has not done some adventuring! It is exactly like the original "Colossal Cave" program I remember. If you have floppies, it becomes a little slow at times (even with 6ms stepping rate drives). I would highly recommend using a hard disk with this program (although I would not recommend buying a hard disk just for the purpose of playing this adventure a little faster). The disk is self documented for instructions on its use. You have to make sure that the data directory MISC is in your current root directory. Unlike Disk #6, this program only requires a single sided 5 1/4" disk.

Application for membership in the OS-9 Users Group

Last: _____

First: _____

Middle: _____

Street: _____

City: _____

State: _____ ZIP: _____

Phone: (_____) _____

Delphi ID: _____

Computer Make and Model: _____

OS-9 Level: _____ (I) _____ (II) _____ (68K) _____ (CoCo)

Disk Size: _____ (5 1/4") _____ (8")

Disk Format: _____ (CoCo) _____ (Standard)

Single Sided: _____ Double Sided: _____

of tracks: _____ (35) _____ (40) _____ (80)

Other restrictions, formats, comments, etc: _____

Enclose \$25 to cover the first year's membership in the form of a personal check or money order. Please allow approximately 3-5 weeks for processing your membership. The Users Group has a small staff, and applications are averaging 30 to 50 a week.

Shortly after acceptance of your application for membership, you will receive the current Group newsletter ("MOTD"), and soon after, the "starter" diskette, UG Disk # 0, with software of the type useful in getting you started with both OS-9 and the Users group, including a modem program to assist you with CompuServe access while under OS-9. Additional diskettes may be purchased at a cost of \$5 each to cover the cost of media and postage.

Mail your application and other correspondence to:

The OS-9 User's Group
ATTN: (department)
9743 University Avenue
Suite 330
Des Moines, IA 50322

Where (department) is President, MOTD Editor, Librarian, Membership, etc.

Membership dues can be charged to your VISA or MasterCard. Simply enter your card information below and sign this form.

Card type: VISA _____ MasterCard _____

Account #: _____

Expires: _____

Is this a Renewal? Yes _____ No _____

OS-9 Has Heros Too!

By Dale L. Puckett
Director-at-Large

Publications — even technical newsletters like MOTD — serve people. And people — even hackers — are interested in people. Following this logic, we quickly realize that if we want you to read and enjoy MOTD, we need to write about people occasionally. This month we'll do just that as we salute a few OS-9 folk heroes.

Some people merely use OS-9 to run application programs. To them, our favorite operating system is only a means to an end — a tool that helps them get their job done. There is nothing wrong with this approach. In fact without these people, there would be no need for OS-9 at all.

Yet, others serve our community above and beyond the call of duty. They spend untold hours designing and writing tools that help others do their job. Some are spending the time because the tools are needed — others are performing a labor of love. In any case, we all benefit.

The heroes of whom we speak this month come from the Color Computer Community. They are the people who are making our evolution from Microware's OS-9 Level 1, Version 1.01 to Version 2.00.00 less painful.

Change almost always creates hassles and the world of personal computer software is not immune. That's why in some ways our current evolutionary process has not been painless.

Excellent software like OS-9 is well documented and generally, if you follow a fixed set of rules, you can guarantee that your software will run on new versions of an operating system. But, sometimes the improvements required by a new version make it impossible to maintain 100 percent compatibility with an older version. We often have to give up a little to gain a lot. That's where our OS-9 heroes enter the picture.

Who then, are our heroes? Actually, there are several and most have provided

third party support to the Color Computer versions of OS-9 since the beginning. Before we name them we want to take time to salute the members of Ken and Jeanne Kaplan's Color Computer OS-9 Team at Des Moines. Mark Hawkins, Tim Harris and Todd Earles did nothing less than an outstanding job.

When Tandy decided that they wanted Microware to do a new version of OS-9 early in 1985 these guys really dug in. They were thinking way past the Color Computer and viewed the job as an opportunity to introduce a brand new version of OS-9 for the 6809. Tandy Version 2.00.00 is the results.

When they were finished, Mark, Tim and Todd had taken care of most of our complaints and had added a handful of utilities that really made life nice. Beginners will now find their transition into OS-9 much easier. Oldtimers however — especially those of us who have added third party hardware to our trusty Color Computers — were in for a bumpier ride. The team's software fixes to hardware induced problems interfered with our third party hardware in several cases.

For example, one of the major problems surrounding the use of hardware external to the Color Computer revolves around the design of the Radio Shack expansion interface. Since OS-9 uses interrupts whenever possible, it needs to be able to receive interrupt signals directly from the hardware. With the expansion interface, this is only possible from one slot. Hardware plugged into other slots can not send an interrupt to the 6809.

To get around this problem, Microware's design team added a virtual interrupt, or VIRQ, capability to Color Computer OS-9. It improved the performance of the system immensely. But since VIRQs didn't exist until Version 2.00.00, the third party software drivers for third party hardware had to be redone.

Additional problems came from the limited memory of the Color Computer. As Mark Hawkins said at the Microware Seminar in Des Moines, "every time you add a new feature to an operating system, you add a couple of lines of code."

The price we paid in the Version 2.00.00 includes a complete redesign of CCIO, the device driver that talks to the Color Computer keyboard and screen. To make more memory available to high level language compilers like C and Pascal, the design team split CCIO into several parts. Only the parts that are needed are in memory at any one time. It works great and the compilers work better. But again we had to pay the price of inconvenience to third party hardware and software suppliers.

Yet, just as they have since OS-9 was made available on the Color Computer, several people stepped forward and solved the problems. And all things considered, they have done a tremendous job.

Ed Bender at PBJ is our first hero. Ed has handled the software chores for Al Alberto and the gang at PBJ since they first introduced the WordPak 80-column card several years ago. Since then he has written drivers for a new WordPak, a parallel port, a dual serial port and a hardware real time clock. All went well for several years — then came Version 2.00.00.

Users were shocked to find that WordPak didn't work with Version 2.00.00. Once you use an 80-column card, you don't like to go back to that 32-column screen. The phones in New Jersey rang off the hook.

Ed dug out his Dynamite disassembler and went to work. After several weeks of hard study he broke the code and got his new drivers on-line. He received little or no support from the software folks at Tandy, which strikes me strange when

you consider that they now sell PBJ's WordPak in their Express Order catalog.

But, when Bender got the WordPak drivers on line, the job wasn't finished. He still had to re-do the clock drivers and the parallel port drivers. Fortunately, since PBJ's 2SP dual serial port uses the same chip as the RS232 cartridge from Tandy, users could use the standard Color Computer /T2 device descriptor and Ed didn't have to rewrite it. Ed Bender, we salute you.

Moving west to Portland, Oregon, we meet our second hero — Dan Johnson of D. P. Johnson. Dan's fame in the OS-9 world comes from his introduction of his SDisk drivers soon after the Color Computer version of OSP9 was introduced.

CCDisk, the original device driver from Tandy, and the device descriptors that used it would only recognize a "standard" Color Computer OS-9 disk. This standard disk was single sided and only had 35 tracks. Dan came to the rescue and at last we could all read and write single and double sided as well as single, double and quad density disks with Color Computer OSP9. And, we could read and write both Radio Shack "standard" OS-9 disks and the "standard" disk written by all other OS-9 based computers. Happy Days were here again — thanks to Dan Johnson. Salute!

Our last hero of the month is Paul Seaby at Computerware in California. We salute Paul for the tremendous speech he delivered at the OSP9 breakfast at Rainbowfest Palo Alto. But more importantly, we salute him for the continuous support he has given the Color Computer OS-9 community.

Paul too, released a replacement for Tandy's CCDisk driver early in the game. It worked and worked well. And, Paul too had to rewrite that driver with the birth of Version 2.00.00. But rewrite it he did and he had it on the streets within a few weeks of the time the official Tandy disks reached the West Coast. The new version also works and works well. But only because of the tremendous effort poured into it by Paul and his software engineers.

We also salute Paul and the rest of the gang at Computerware for their innovation. Their Disk Fix, Utilities and Advanced Utilities packages really streamline our OS-9 operation. An example is the new KShell which comes in the new Advanced Utilities package. It emulates the Shell in OS-9 68K as closely as possible — hence its name. The addition of wild cards, macros and paths all adds up to smoother and more effective development.

Dan, Ed and Paul — we salute you all. Thanks for taking the time and effort and for going that extra mile to make your products compatible with Version 2.00.00 so quickly.

And to our readers — the next time you order software from Dan, Ed and Paul or the many other third party suppliers who make OS-9 shine on the Color Computer — make sure you thank them personally. Let them know that you appreciate what they have done for us all.

But most importantly, let others know that you appreciate what our heroes had to go through. Tell them you hope that when future enhancements of Color Computer OS-9 come on-line our heroes will have had the benefit of an early release — or at least a very specific description that will let them prepare for the rush — before the new release hits the streets. You really shouldn't have to stay up all night disassembling someone else's code so you can help expand the use of their product.

If you have other heroes who you would like to see saluted in MOTD, send us their names. Till then, keep on hacking!

LLOYD I/O

Lloyd I/O is a computer engineering corporation providing software and hardware products and consulting services.

19535 NE GLISAN • PORTLAND, OR 97230 (USA)
PHONE: (503) 666-1097 • TELEX: 910 380 5448 LLOYD I/O

New Product!

CRASMB™ CROSS ASSEMBLER NOW AVAILABLE FOR OS9/68000

LLOYD I/O announces the release of the CRASMB 8 Bit Macro Cross Assembler for Microware's OS9 disk operating system for the 68000 family of microprocessors. In recent increasing demand for the OS9/68000 version of CRASMB, LLOYD I/O has translated its four year old CRASMB for the OS9/6809 and FLEX/6809 to the OS9/68000 environment.

CRASMB supports assembly language software development for these microprocessors: 1802, 6502, 6800, 6801, 6303, 6804, 6805, 6809, 6811, TMS 7000, 8048/family, 8051/family, 8080/85, Z8, and the Z80. CRASMB is a full featured assembler with macro and conditional assembly facilities. It generates object code using 4 different formats: none, FLEX, Motorola S1-S9, and Intel Hex. Another format is available which outputs the source code after macro expansion, etc. CRASMB allows label (symbols) length to 30 characters and has label cross referencing options.

CRASMB for OS9/68000 is available for \$432 in US funds only. It may be purchased with VISA/MASTERCARD cards, checks, US money orders, or US government (federal, state, etc.) purchase orders. NOTE: please add \$5 shipping in the USA and use your street address for UPS shipments. Add \$30 for all overseas orders. CRASMB for OS9/6809 and FLEX/6809 cost \$399 plus shipping.

You may contact Frank Hoffman at LLOYD I/O, 19535 NE Glisan, Portland, Oregon, 97230. Phone: (503) 666-1097. Telex: 910 380 5448, answer back: LLOYD I/O. Easylink: 62846110. See list of distributors below.

VISA, MC, C.O.D. CHECKS ACCEPTED
USA: LLOYD I/O (503 666 1097) S.E. MEDIA (800 338 6800)
England: Vivaway (0562 423425) Windrush (0692 405189)
Germany: Zacher Computer (65 25 299) Kell Software (06203 6741)
Australia: Parts Radio Electronics (344 9111)
Japan: Microboards (0474) 22-1741 Seikou (03) 832-6000
Switzerland: Elsoft AG (056 86 27 24)
Sweden: Micromaster Scandinavian AG (018 - 138595)

* BASIC, DO, SEARCH and RESCUE UTILITIES
PATCH, CRASMB and CRASMB to 32 are trademarks of LLOYD I/O
OS9 is a trademark of Microware. FLEX is a trademark of TSC

DESKMATE

a review

by Bruce N. Warner

Have you ever wondered what Color Computer owners had that other OS-9 users were missing? Then again, have you ever wondered what Tandy's MS-DOS users have that we Color Computer Owners are missing? The first answer is Color Graphics, and the answer to the second is DESKMATE. After more than two years of working with MS-DOS, I'm sold on the fact that there's nothing you can do in MS-DOS that can't be done better under OS-9.

Just recently, Tandy decided to put both of these features together. In doing so, they've come up with a whole new package for the Color Computer that is in the tradition of Tandy's dedication to OS-9. The program is DESKMATE, and it's a real bonus to any OS-9 user.

Until I got into the text editor, I found it necessary to read only one page of the DESKMATE manual. That was the one that told me how to switch from using the arrow keys to using a joystick (you start by using the Alt key, a new feature for Version 02.00.00). After that, I found that there are enough windowed help menus and prompts to make the remainder of using DESKMATE require almost no use of the manual. Everything from the appointment calendar (displayed in the upper left corner of DESKMATE's main menu) down to the Text Editor (shown in the lower right corner) uses the same addressing schemes. Help menus are available throughout.

DESKMATE comes with six functions. In addition to the two mentioned in the previous paragraph, there are: Ledger, Telecom, Index Cards and Paint (similar, but not as good as CoCo Max). We'll discuss each of these functions separately as well as their required hardware additions for complete use.

Calendar

If you've wanted a computer calendar that gives you complete access to any date, a display of the month the day falls in, a search by date or event (after programming a date, look for a key word in that event) and a printed copy of the calendar, DESKMATE is the program

for you. All you need is a printer to use all of the features of calendar. It works flawlessly.

Start with opening the file and search for a starting date. After that, you need do nothing more than follow the prompts. The entire month you're looking at will be displayed with the specific date in reverse type. If you should choose to edit text for the day in question, you'll be delighted to know that it's very easy to use and requires little more than pressing the keys on your keyboard.

Calendar alone is worth at least 40% of the cost of the package (we'll add up the percentages later if Tandy promises not to raise the price).

Ledger

If you found Color Disk Spectaculator to be a functional program, you'll be glad to see it as a part of DESKMATE. It's not exactly the same as Spectaculator, but it's close enough to make you see the similarities in capabilities and limitations. The big difference is that the DESKMATE Ledger can address individual cells and it is part of a complete package.

Telcom

The DESKMATE communications package requires the Deluxe RS-232 pack, a multi-pak interface and a modem. On my system I use a SMARTTEAM 1200 Baud Hayes compatible modem. The package works well and with the strip utility it is (run in this issue of the MOTD) it's a dream. My one disappointment is that the graphics screen slows down the data transfers.

Index cards

What makes a database so scary to the novice is the idea of what a database is. When I called up the index card function of DESKMATE, I was shocked to see that it is a small database. It has most of the functions needed by home computer users and it's in a format they understand. It doesn't have all of the integration functions, nor does it have all of the

we'll help take the bite out of contributing.

So what do you write about? Anything you're doing. I've been asked if it's a good idea to write about interfacing external hardware devices and robot control. Another was about their solution to a BASIC09 problem. Another was about a problem with a vendor. The answer is yes, yes, yes, yes and YES! If it's something other UG members should know about, it's something that needs to be told in the MOTD. That's what a users group newsletter is for and about.

No one can say that I didn't warn you that this was coming. The dead horse is being beaten to death again, and I'll continue to do so until there's no room in my house for the excess copy!!!

fancy stuff we programmers like, but it's written for everyday users, not hackers.

Paint

They say a picture's worth a thousand words. The Paint utility of DESKMATE isn't the fanciest thing I've ever seen, but it sure gets the job done, and under OS-9. This utility is only lacking in the fact that we true OS-9ers aren't into a lot of graphics. That's not to say that I didn't have fun playing with Paint, it's just not a requirement for my programming needs. The one nice thing is that it does its job well.

Overview

There are two ways to look at the DESKMATE package. You can look at it like a true hacker, in which case it lacks a lot, or you can look at it like a typical Color Computer OS-9 user, in which case, it's a nice, friendly package that gives you all you need to use your Color Computer in some meaningful fashion.

My recommendation is that if you are a heavy OS-9 user, DESKMATE will fall short of your needs, but if you're a newer user, DESKMATE may be all the OS-9 software you'll ever need.

OS-9 Users Group Software Library Volumes — 01/29/86

No.	Done?	Title:	Format:	Who?
0.05	Y	New Member Intro	+++ (40 track, ss)	DK
1.00	Y	Spelling Checker	(35 track, ss)	DK
2.00	Y	Spelling Dictionary	(40 track, ds)	DK
3.01	Y	Word Processing Utils	+++ (35 track, ss)	DK
4.01	Y	Programming Utilities	+++ (35 track, ss)	DK
5.00	Y	File Processing Utils	+++ (35 track, ss)	DK
6.02	Y	Adventure Game (source)	(40 track, ds)	CK
7.02	Y	Adventure Game (object)	(40 track, ss)	CK
8.00	Y	General Interest (demo, games, finance)	(35 track, ss)	CK
9.00	Y	C Programmer's Tool Kit	(35 track, ss)	CK
10.00	Y	Math & Electronics I	(35 track, ss)	GD
11.00	Y	Word Processing Utils (disk #2)	(35 track, ss)	CK
12.00	Y	Programming Utilities (disk #2)	(35 track, ss)	CK
13.00	Y	File Processing Utils (disk #2)	(35 track, ss)	CK
14.02	Y	File Maintenance	(35 track, ss)	CK
15.01	Y	Communication	(35 track, ss)	CK
16.00	Y	Hardware Customizations	(35 track, ss)	CK
17.00	Y	BASIC09 Programmer's Tool Kit	(35 track, ss)	CK
18.00	Y	System Utilities	(35 track, ss)	CK
19.01	Y	Languages I: XLisp (source)	(40 track, ds)	CK
20.00	Y	XLisp (object)	(35 track, ss)	DK
21.00	Y	File maintenance (disk #2)	+++ (35 track, ss)	CK
22.00	Y	Programming Utilities (disk #3)	(35 track, ss)	CK
23.00	Y	File Processing Utils (disk #3)	(35 track, ss)	CK
24.00	Y	General Interest (disk #2)	(35 track, ss)	CK
25.00	Y	Word Processing Utils (disk #3)	(35 track, ss)	CK
26.00	Y	C Language Math Library	(35 track, ss)	CK
29.00	Y	File Maintenance (disk #3)	(35 track, ss)	CK
30.00	Y	File Processing Utils (disk #4)	(35 track, ss)	CK
31.00	Y	Hardware Customizations (disk #2)	(35 track, ss)	CK
32.00	Y	Hardware Customizations (disk #3)	(35 track, ss)	CK
33.00	Y	System Utilities (disk #2)	(35 track, ss)	CK
34.00	Y	Hardware Customizations (disk #4)	(35 track, ss)	CK
35.00	Y	System Utilities (disk #3)	(35 track, ss)	CK
36.00	Y	General Interest (disk #3)	(35 track, ss)	CK
37.00	Y	Communication (disk #2)	(40 track, ds)	CK
38.00	Y	Programming Utilities (disk #4)	(35 track, ss)	CK
39.00	Y	Communication (Morse) (disk #3)	(40 track, ds)	CK
40.00	Y	System Utilities (disk #4)	(35 track, ss)	CK
42.00	Y	Coco Graphics	(35 track, ss)	CK

NOTES: 1) Above formats describe STANDARD (non-CoCo) versions and are single-density.

2) All of the completed volumes are available in TRS-80 Color Computer format (dd, 18 sectors/track, etc.)

+++ For these volumes to fit in the specified format, the disk is created with a default sector allocation of 1 sector per directory (made by doctoring the "segment allocation size" byte (offset \$20) in the device descriptor of the drive on which the master disk is made).

Orders can be made by mail or through the online services of DELPHI and CompuServe. Payment is required in advance by check, money order, VISA or Master Card.

Last Name _____ First _____ Initial _____

Street Address _____

City _____ State _____ Zip _____

OS-9 Disk Format _____ Form of payment _____

Disk No.	Qty.	Cost	Amount
		\$5.00	
		\$5.00	
		\$5.00	
		\$5.00	
5" Archive Set		\$70.00	

If paying by credit card enter the following:

Card type: VISA _____ MasterCard _____

Account #: _____ Exp. Date _____

Signature _____

WRITING FOR THE MOTD

Have you ever wanted to be an internationally known author? I remember as a kid (last week) wanting to be Superman, but knowing I'd never be able to bounce bullets off my chest I decided to settle for writing (Clark Kent).

You too can be a mild-mannered reporter for a growing international newsletter. It's easy! Just put down on disk what you're going through with your favorite OS-9 computer, and send the disk, along with a paper copy, to the Editor of the MOTD.

By the end of the year we'll be setting up to reimburse you for the disk and paper. As a users group, we think every member should contribute something to the group. Though we aren't paying for the submissions themselves, as the MOTD becomes more self-sufficient,

Continued from page 4

```

00092 01CE 1F20          tfr  Y,D          move buffer size to D
00093 01D0 31C90250     leay BUFBEQ,U    get buffer address
00094 01D4 3420          pshs Y           add size to beginning address
00095 01D6 E3E1          addd ,S++        /to calculate beginning output
00096 01D8 DD02          std  BUFPTR      save output buffer address
00097 01DA 0F07          clr  MINUSC      "-c" option flag
00098 01DC 0F08          clr  MINUSL      "-l" option flag
00099 01DE 0F09          clr  PLUSL       "+l" option flag
00100 01E0 0FOA          clr  PLUSB       "+b" option flag
00101 01E2 0FOB          clr  PLUSD       "+d" option flag
00102 01E4 0F06          clr  ORDER       clear parameter order counter
00103 01E6 0F0C          clr  FLAG        clear flag
00104 01E8 0FOF          clr  STRPCNT     clear strip count
00105 01EA 0F10          clr  STRPCNT+1  /
00106          *****
00107          * Parse for options
00108          *
00109 01EC A680          STRIP02 ldd  ,X+         Parse parameters
00110 01EE 1F89          tfr  A,B         save in both A and B
00111 01F0 812D          cmpa #'-         Start of minus parameter?
00112 01F2 2712          beq  STRIP03     yes process it
00113 01F4 812B          cmpa #'+         Start of plus parameter?
00114 01F6 2730          beq  STRIP05     Yes, go process it.
00115 01F8 813F          cmpa #'?         Help message request?
00116 01FA 275E          beq  STRIP06     Yes, go process it.
00117 01FC C10D          STRIP07 cmpb #C$CR      End of line?
00118 01FE 276A          beq  STRIP20     no, keep looking
00119 0200 C120          cmpb #C$Spac    delimiter?
00120 0202 27E8          beq  STRIP02     yes, continue
00121 0204 2064          bra  STRIP20     No more options
00122          *****
00123          * Handle "-c or -l" options
00124          *
00125 0206 EC84          STRIP03 ldd  ,X          get the option char
00126 0208 845F          anda #%01011111 mask off lower case
00127 020A 8143          cmpa #C$ASCC    C option?
00128 020C 260A          bne  STRIP04     ..no, not this option
00129 020E 0C06          inc  ORDER       increment order counter
00130 0210 9606          lda  ORDER       get order counter
00131 0212 9707          sta  MINUSC     set carriage return flag
00132 0214 3002          leax 2,x        adjust parameter pointer
00133 0216 20E4          bra  STRIP07     Look for more parameters
00134 0218 EC81          STRIP04 ldd  ,X++        get option char
00135 021A 845F          anda #%01011111 mask off lower case
00136 021C 814C          cmpa #C$ASCL    L option?
00137 021E 263A          bne  STRIP06     ..no, not this option
00138 0220 0C06          inc  ORDER       increment order counter
00139 0222 9606          lda  ORDER       get order counter
00140 0224 9708          sta  MINUSL     set line feed flag
00141 0226 20D4          bra  STRIP07     Look for more parameters
00142          *****
00143          * Handle "+l , +b or +d" options
00144          *
00145 0228 EC84          STRIP05 ldd  ,X          Get the option character
00146 022A 845F          anda #%01011111 mask off lower case
00147 022C 814C          cmpa #C$ASCL    L option?
00148 022E 260A          bne  STRIP08     ..no, not this option
00149 0230 0C06          inc  ORDER       increment order counter
00150 0232 9606          lda  ORDER       get order counter
00151 0234 9709          sta  PLUSL      set line feed flag
00152 0236 3002          leax 2,X        adjust parameter pointer
00153 0238 20C2          bra  STRIP07     Look for more parameters
00154 023A EC84          STRIP08 ldd  ,X          get option char
00155 023C 845F          anda #%01011111 mask off lower case
00156 023E 8142          cmpa #C$ASCB    B option?
00157 0240 260A          bne  STRIP09     ..no, not this option
00158 0242 0C06          inc  ORDER       increment order counter
00159 0244 9606          lda  ORDER       get order counter
00160 0246 970A          sta  PLUSB      set back space flag
00161 0248 3002          leax 2,X        adjust parameter pointer
00162 024A 20B0          bra  STRIP07     Look for more options
00163 024C EC81          STRIP09 ldd  ,X++        get option char
00164 024E 845F          anda #%01011111 mask off lower case
00165 0250 8144          cmpa #C$ASCD    D option?
00166 0252 2606          bne  STRIP06     option not found
00167 0254 86FF          lda  #$FF       get order counter
00168 0256 970B          sta  PLUSD      set display flag
00169 0258 20A2          bra  STRIP07     Look for more options
00170          *****
00171          * Give help message on error

```

```

00172      *
00173 025A 8602      STRIP06 lda #2      setup for standard error outpu
00174 025C 308DFDB3 leax STRPUSE,PCR get help message location
00175 0260 108E0169 ldy #STRPSIZ get help message size
00176 0264 103F8C OS9 I$WritLn give message
00177 0267 160165 lbra STRIP98 Exit
00178      *****
00179      * Do the strip
00180 026A 30C90250 STRIP20 leax BUFBEg,U Get the buffer address
00181 026E 109E00 ldy BUFSIZ Get the buffer size
00182 0271 8600 lda #0 Get standard input path
00183 0273 103F89 OS9 I$Read Read from standard input
00184 0276 2409 bcc STRIP25 Encounter an error during READ
00185 0278 C1D3 cmpb #E$EOF Yes. Was it an EOF?
00186 027A 10270136 lbeq STRIP97 Exit
00187 027E 16014F lbra STRIP99 No. Exit with error code.
00188 0281 3420 STRIP25 pshs Y save transfer size
00189 0283 1F10 tfr X,D move input buffer pointer to D
00190 0285 E3E1 addd ,S++ add input buffer pointer to re
00191 0287 DD04 std BYTCNT Update the buffer size count.
00192 0289 109E02 ldy BUFPTR get output buffer pointer
00193 028C A680 STRIP26 lda ,X+ get character
00194 028E 847F anda #%01111111 mask off bit 7
00195 0290 8120 cmpa #C$Spac compare to space
00196 0292 2437 bhs STRIP27 branch if not control code
00197 0294 810D cmpa #C$CR is character a carriage return
00198 0296 260D bne STRIP36 ..no, branch
00199 0298 0D07 tst MINUSC minus carriage return option?
00200 029A 274A beq STRIP29 no, skip
00201 029C 1700A1 lbrs STRIP39 display character
00202 029F 1025012D lbrs STRIP99 branch if error
00203 02A3 2041 bra STRIP29 continue
00204 02A5 810A STRIP36 cmpa #C$LF is character a line feed?
00205 02A7 260D bne STRIP37 ..no, branch
00206 02A9 0D08 tst MINUSL minus line feed option
00207 02AB 2739 beq STRIP29 no, skip
00208 02AD 170090 lbrs STRIP39 display character
00209 02B0 1025011C lbrs STRIP99 branch if error
00210 02B4 2030 bra STRIP29 continue
00211 02B6 8108 STRIP37 cmpa #C$BS is character a back space?
00212 02B8 170083 lbrs STRIP39 ..no, branch
00213 02BA 170083 lbrs STRIP39 display character
00214 02BD 1025010F lbrs STRIP99 branch if error
00215 02C1 2025 bra STRIP30 continue
00216 02C3 8D7B STRIP38 bsr STRIP39 display character
00217 02C5 10250107 lbrs STRIP99 branch if error
00218 02C9 2002 bra STRIP28 continue
00219 02CB A7A0 STRIP27 sta ,Y+ save character in output buffe
00220 02CD 9C04 STRIP28 cmpx BYTCNT last character in input buffer
00221 02CF 26BB bne STRIP26 no look for more characters
00222 02D1 9E02 ldx BUFPTR get output buffer pointer
00223 02D3 3410 pshs X save on stack
00224 02D5 1F20 tfr Y,D transfer output buffer pointer
00225 02D7 A3E1 subd ,S++ claculate output buffer size
00226 02D9 1F02 tfr D,Y transfer size to Y
00227 02DB 8601 lda #1 setup for standard output
00228 02DD 103F8A OS9 I$Write write output buffer
00229 02E0 102500EC lbrs STRIP99 branch if error
00230 02E4 2084 bra STRIP20 get next buffer
00231 02E6 A7A0 STRIP29 sta ,Y+ save carriage return or line f
00232 02E8 0F06 STRIP30 clr ORDER clear parameter order counter
00233 02EA 0C06 STRIP31 inc ORDER increment order counter
00234 02EC D606 ldb ORDER get order counter
00235 02EE D107 cmpb MINUSC C option next?
00236 02F0 270E beq STRIPC ..yes, branch
00237 02F2 D108 cmpb MINUSL L option next?
00238 02F4 2724 beq STRIPL ..yes, branch
00239 02F6 D109 cmpb PLUSL L option next?
00240 02F8 2726 beq ADDL ..yes, branch
00241 02FA D10A cmpb PLUSB B option next?
00242 02FC 272C beq ADDB ..yes, branch
00243 02FE 20CD bra STRIP28 ..done, get next character
00244 0300 810D STRIPC cmpa #C$CR is character a carriage return
00245 0302 272A beq ADDB1 ..yes, adjust output buffer po
00246 0304 810A cmpa #C$LF is character a line feed?
00247 0306 26E2 bne STRIP31 ..no, get next option
00248 0308 313E leay -2,Y get charcter before line feed
00249 030A A6A4 lda ,Y /
00250 030C 810D cmpa #C$CR is character a carriage return
00251 030E 2704 beq STRIPC1 ..yes, strip carriage return
00252 0310 3122 leay 2,Y adjust output buffer pointer
00253 0312 20D6 bra STRIP31 get next option
00254 0314 860A STRIPC1 lda #C$LF save line feed over carriage r

```

00255	0316 A7A0		sta	,Y+	/
00256	0318 20D0		bra	STRIP31	get next option
00257	031A 810A	STRIPL	cmpa	#C\$LF	is character a line feed?
00258	031C 26CC		bne	STRIP31	..no, get next option
00259	031E 200E		bra	ADDB1	..yes, adjust output buffer po
00260	0320 810D	ADDL	cmpa	#C\$CR	is character a carriage return
00261	0322 26C6		bne	STRIP31	..no, get next option
00262	0324 860A		lda	#C\$LF	get line feed
00263	0326 A7A0		sta	,Y+	save in output buffer
00264	0328 20C0		bra	STRIP31	get next option
00265	032A 8108	ADDB	cmpa	#C\$BS	is character a back space?
00266	032C 26BC		bne	STRIP31	..no, get next option
00267	032E 109C02	ADDB1	cmpy	BUFPtr	is output buffer pointer at be
00268	0331 27B7		beq	STRIP31	..yes, get next option
00269	0333 313F		leay	-1,Y	back up output buffer pointer
00270	0335 A6A4		lda	,Y	get deleted character
00271	0337 8D07		bsr	STRIP39	display character
00272	0339 10250093		lbcs	STRIP99	branch if error
00273	033D 4F		clra		clear character
00274	033E 20AA		bra	STRIP31	get next option
00275	0340 3432	STRIP39	pshs	A,X,Y	save character
00276	0342 D610		ldb	STRPCNT+1	get strip count
00277	0344 CB01		addb	#1	increment
00278	0346 D710		stb	STRPCNT+1	restore
00279	0348 D60F		ldb	STRPCNT	get second byte
00280	034A C900		adcb	#0	add carry from first byte
00281	034C D70F		stb	STRPCNT	restore
00282	034E 0D0B		tst	PLUSD	display stripped characters?
00283	0350 2734		beq	STRIP41	no, skip
00284	0352 0D0C		tst	FLAG	first character?
00285	0354 2619		bne	STRIP40	no, skip
00286	0356 3402		pshs	A	save registers
00287	0358 8602		lda	#2	setup for error output
00288	035A 970C		sta	FLAG	set flag
00289	035C 308DFE1C		leax	STRIPD,PCR	get 'stripped' message
00290	0360 108E0017		ldy	#STRPDSZ	get message size
00291	0364 103F8C		OS9	I\$WritLn	give message
00292	0367 3502		puls	A	restore registers
00293	0369 251D		bcs	STRIP42	branch if error
00294	036B 8D1E		bsr	DISP	display character
00295	036D 2019		bra	STRIP42	continue
00296	036F 3402	STRIP40	pshs	A	save registers
00297	0371 8602		lda	#2	setup for error output
00298	0373 308DFE38		leax	SEP,PCR	get ',' message
00299	0377 108E0002		ldy	#SEPSZ	get message size
00300	037B 103F8A		OS9	I\$Write	give message
00301	037E 3502		puls	A	get registers
00302	0380 2506		bcs	STRIP42	branch if error
00303	0382 8D07		bsr	DISP	display character
00304	0384 2002		bra	STRIP42	exit with error
00305	0386 1CFE	STRIP41	andcc	##11111110	clear error flag
00306	0388 3532	STRIP42	puls	A,X,Y	restore character
00307	038A 39		rts		
00308	038B 1F89	DISP	tfr	A,B	save character
00309	038D 44		lsra		right justify character
00310	038E 44		lsra		/
00311	038F 44		lsra		/
00312	0390 44		lsra		/
00313	0391 8109		cmpa	#9	character over 9?
00314	0393 2302		bis	DISP1	no, skip
00315	0395 8B07		adda	#7	add offset
00316	0397 8B30	DISP1	adda	#\$30	convert to ASCII
00317	0399 970D		sta	CHAR	save character in output strin
00318	039B C40F		andb	##00001111	get lower nibble
00319	039D C109		cmpb	#9	character over 9?
00320	039F 2302		bis	DISP2	no,skip
00321	03A1 CB07		addb	#7	add offset
00322	03A3 CB30	DISP2	addb	#\$30	convert to ASCII
00323	03A5 D70E		stb	CHAR+1	save character in output strin
00324	03A7 8E000D		ldx	#CHAR	get character string
00325	03AA 108E0002		ldy	#2	setup for 2 characters
00326	03AE 8602		lda	#2	setup for error output
00327	03B0 103F8A		OS9	I\$Write	write string
00328	03B3 39		rts		
00329	*****				
00330	* Exit				
00331	*				
00332	03B4 8602	STRIP97	lda	#2	setup for error output
00333	03B6 308DFDD9		leax	TERM,PCR	get Terminate message
00334	03BA 108E001C		ldy	#TERMSIZ	get message size
00335	03BE 103F8C		OS9	I\$WritLn	
00336	03C1 250D		bcs	STRIP99	
00337	03C3 960F		lda	STRPCNT	

MOTD SEPTEMBER 1986
PAGE 8
L 2 AX CHAR, 4 →
MOTD MAY/JUNE 1987 P11

```

00338 03C5 8DC4      bsr  DISP
00339 03C7 2507      bcs  STRIP99
00340 03C9 9610      lda  STRPCNT+1
00341 03CB 8DBE      bsr  DISP
00342 03CD 2501      bcs  STRIP99
00343 03CF 5F         STRIP98  clrb  Return no errors
00344 03D0 103F06    STRIP99  OS9  F$Exit  Terminate
00345 03D3 43D640    emod  Module CRC
00346 03D6         STPEND  equ  *
00347         end

```

```

00000 error(s)
00000 warning(s)
$03D6 00982 program bytes generated
$1000 04096 data bytes allocated
$26A4 09892 bytes used for symbols

```

WHAT'S NEW FROM DES MOINES?

A look at the CompuServe OS-9 conference call with Ken Kaplan

COPYRIGHT 1985 - THE COMPUSERVE OS9 SIG. REPRINTED BY PERMISSION.

THE FOLLOWING IS AN EDITED TRANSCRIPT OF PARTS OF THE OS9 SIG'S CONFERENCE WITH MICROWARE, THE INVENTORS OF OS9. SPEAKING FOR "MICROWARE" IS KEN KAPLAN, PRESIDENT, AND ANDY BALL, VICE PRESIDENT OF MARKETING.

OUR GUESTS, ARE KEN KAPLAN, PRESIDENT OF MICROWARE, THE INVENTORS OF OS9, AND HIS VP OF MARKETING, ANDY BALL. LET'S START BY LETTING KEN AND ANDY SAY ANYTHING (LEGAL HE HE) THAT THEY WISH. KEN OR ANDY? (GA)

MICROWARE HI EVERYONE!

WELL, THAT WAS CERTAINLY ILLUMINATING! (HE HE) KEN, TO STAVE OFF A FEW QUESTIONS FROM PEOPLE NOT FAMILIAR WITH OS9, WOULD YOU CARE TO GIVE US A BRIEF HISTORY OF OS9, AND A BRIEF OUTLINE OF JUST WHAT THE HECK OS9 IS? MICROWARE IN 10,000 WORDS OR LESS (HE HE)?

WELL, I KNOW MANY WILL ASK THOSE QUESTIONS, SO IT MAY SAVE TIME

HOW ABOUT 50 WORDS?

MICROWARE OK. OS-9 WAS ORIGINALLY WRITTEN IN 1979 FOR MOTOROLA FOR THE 6809 MPU. IN 1980 WE STARTED ON A 68K VERSION OS-9 THAT HAS BECOME WIDELY USED IN A GREAT VARIETY OF APPLICATIONS BECAUSE IT'S SMALL, FAST, REAL TIME, MULTITASKING, ROMABLE, ETC.

BOB M. OK . . . WE HEAR A LOT ABOUT OS-9 LEVEL 2 AND I WONDERED JUST WHAT IS DIFF. BTW LEVEL 1 AND 2. I KNOW LEVEL 2 WITH THE RIGHT HARDWARE CAN ACCESS A LOTTTT OF RAM. BUT IS THIS>>> A LEGAL COMMAND UNDER OS-9 LEVEL 2.

```
OS9:EDIT /D0/MYFILE
#120K???? (GA)
```

MICROWARE EACH PROCESS IS LIMITED TO 64K - BUT YOU CAN HAVE MORE MEMORY MODULES PRE-LOADED IN RAM WHICH CAN BE QUICKLY LINKED AND UNLINKED. 64K - CAN'T GET AROUND A 16 BIT PC REGISTER IN SOFTWARE (GA)

GEORGE B. KEN, OS-9 AND OS-9/68000 IS THE GREATEST THING SINCE SLICED BREAD BUT THE LANGAUGE COMPILERS ARE SO BLOODY EXPENSIVE THAT US "HACKER" CAN'T AFFORD TO GET THEM IS THERE CHANCE OF SOME KIND

OF PRICE BREAK FOR "PERSONAL" USE COMPILERS?? (GA)

MICROWARE THAT'S A PROBLEM BECAUSE MOST OF OUR 68K CUSTOMERS ARE INDUSTRIAL AND THE LO-COST PERSONAL COMPUTERS AREN'T OUT YET. BUT I THINK WHEN THEY ARE AVAILABLE THEY'LL BE AFFORDABLE. INCLUDING LANGAUGES. (GA)

GEORGE B. WELL, I SORTA CONSIDER MY UNIQWAD "LOW PRICED" RELATIVELY SPEAKING . . .

DAVE Y. WILL THE ATARI ST VERSION REFLECT TRAMEIL'S PHILOSOPHY OF "POWER WITHOUT THE PRICE", (MEANING LOW COST), AND WHEN WILL IT BE AVAILABLE? (GA)

MICROWARE AFRAID I DON'T KNOW THE EXACT \$\$\$ AND DATE YET, PARTLY BECAUSE ANOTHER COMPANY IS DEVELOPING THIS VERSION. BUT I THINK THERE WILL BE AN OFFICIAL ANNOUNCEMENT IN JANUARY. (GA)

DAVE Y. THANKS. THERE'S ABOUT 50K ST USERS NOW, 100K BY THE END OF THE YEAR (ESTIMATED). THAT'S A POTENTIALLY HIGH MARKET IF THE PROJECTED SALES MATCH ACTUAL.

MICROWARE I'LL ADD TO THAT, DAVE. PRICE WILL BE BETWEEN \$100 AND \$500, AVAILABLE BETWEEN MARCH AND MAY. "C" AND AT LEAST 1 OTHER LANGUAGE WILL BE INCLUDED. SORRY IF THAT'S NOT SPECIFIC ENOUGH, BUT IT'S EARLY IN THE GAME! . . .

MARTIN TORRES THANKS, WILL THE TOWERS IN FT WORTH CONTINUE TO SUPPORT US, WITH OTHER MANUFACTURERS ON LINE TOO AND IF SO, WILL THERE BE CROSS MACHINE COMPATIBILITY? (GA)

MICROWARE PLEASE UNDERSTAND THAT I CAN'T SPEAK FOR THEM OR THEY HAVEN'T ANNOUNCED YET. BUT PERSONALLY SPEAKING, I THINK THAT THE NUMBER OF OS-9 SOFTWARE PACKAGES THEY'VE RELEASED IS AN INDICATION OF VERY STONG SUPPORT. AND I THINK THEY'D BE CRAZY TO ABANDON. ALL THOSE LOYAL COCO OWNERS OUT THERE. AS FOR "CROSS MACHINE COMPATIBILITY" I'VE ALREADY TAKEN BINARIES FROM A GIMIX UNDER LEVEL II, AND DOWNLOADED THEM TO A COCO. WORK PERFECTLY! AND FOR ANOTHER EXAMPLE, YOU'LL BE ABLE TO TAKE A BINARY FROM SAY, AN IBM PC, DOWNLOAD IT TO AN ST OR AMIGA, AND EXECUTE IT IMMEDIATELY. THAT HELP? (GA)

MARTIN TORRES SORRY, AS A USER OF

THE OTHER OS, (THE SLOW ONE THAT BEGINS WITH "M"), A FUTURE WITH OS9/OS9-68K WOULD SURE BE BETTER, TNX AGAIN (END)

MICROWARE THANKS, MARTIN. I THINK YOU'LL BE PLEASANTLY SUPRISED (GA)

CHAS A KEN CAN YOU TELL US ANY THING ABOUT TANDY'S RUMORED LEVEL II MACHINE GA

MICROWARE SORRY. CAN'T CONFIRM OR DENY. I THINK YOU CAN IMAGINE WHY. (GA)

CHAS A OK HOW ABOUT TELLING US THE DIFFERENCES BETWEEN OS9 LEVEL 2 AND OS9/68000 GA

MICROWARE 1. 16 MB MAX MEMORY PER TASK/SYSTEM VS. 2MB/64K . . .

MICROWARE 2. MORE ADVANCED SHELL AND UTILITIES . . . (16MB IS 68000, 2MB IS LEVEL II)

MICROWARE 3. MORE REAL-TIME FUNCTIONS.

MICROWARE 4. MANY OTHER IMPROVEMENTS THROUGHOUT. (GA) (#'S 2, 3, 4 DESCRIBE OS9/68000)

CHAS A KIND OF MAKES YOU WONDER WHY ONE SHOULD WAIT . . .

MICROWARE OS-9/68K IS ESSENTIAL A SUPERSET OF OS-9/6809.

CHAS A FOR A LEVEL II MACHINE FROM TANDY WHEN OTHERS ARE COMMING OUT WITH 68000 MACHIES OUT

THANK YOU, CHAS. THAT'S A DECISION ONLY YOU CAN MAKE. WE DON'T CARE WHAT COMPUTER YOU BUY, AS LONG AS IT RUNS OS9! (HE HE . . . UH OH).

MICROWARE THERE WILL ALWAYS BE A BETTER MOUSETRAP BUT ALL TEND TO CATCH MICE.

CHAS S. I'M CURIOUS ABOUT THE OS9 USER BASE IS OS9 MORE POPULAR IN JAPAN THAN IN THE USA? GA

MICROWARE IN TERMS OF VISIBILTY (BOOKS, ARTICLES, ETC) IT IS MORE SO IN JAPAN BUT IN ABSOLUTE NUMBER THE US IS BIGGER. (GA)

CHAS S. WHAT IS THE PERCENTAGE OF INDUSTRIAL USERS IN THE USA VS. PERSONAL USERS GA (END)

MICROWARE ABOUT 1/3 PERSONAL AND 2/3 INDUSTRIAL. IN JAPAN IT'S ABOUT 50-50. (GA)

JEFF S. KEN, WITH OPTICAL DISK

TECHNOLOGY ON THE HORIZON ARE THERE ANY CAPACITY LIMITATIONS OF OS9/68K ? AND WHAT ARE THE PLANS (IF ANY) FOR VIRTUAL MEMORY IMPLEMENTATIONS? GA

MICROWARE JEFF- (1) NO LIMITATIONS . . . (2) A 68K VM VERSION WAS PLANNED EVENTUALLY, HOWEVER, THE RAPID DROP IN DRAM COST MAY CHANGE THAT. WHY USE A SLOW IMITATION (DISK VM) OF THE REAL THING (RAM) . . . IF THE COST IS ABOUT THE SAME? (GA)

JEFF S. WELL, WE CAN DEBATE THE PRO'S OF VM VS REAL RAM LATER CONCERNING THE COMPATIBILITY OF OS9/68K BETWEEN GRAPHICS ORIENTED MACHINES LIKE THE IBM PC W/68K, ST, AMIGA, ETC . . . IS THERE GOING TO BE A STANDARD GRAPHICS DRIVER PACKAGE FOR OS9/68K ? GA

MICROWARE YES. WE ARE OFFERING THE ISO/ANSI STANDARD VIRTUAL DEVICE INTERFACE (VDI) AS WELL AS A SOON-TO-BE-ANNOUNCED INTERMEDIATE LEVEL DRIVER STANDARD (GA)

JEFF S. I TAKE IT THAT THEY'RE NOT AVAILABLE WITH V 1.2 ? IS THERE POSSIBILITY OF GETTING MACHINE READABLE DOCUMENTATION FOR OS9/68K . . . SINCE IT SEEMS TO BE RAPIDLY BEING ENHANCED ? GA

MICROWARE OUR GRAPHICS SOFTWARE WILL WORK WITH 1.2 BUT THEY'LL PRIMARILY BE SHIPPED W/HARDWARE BECAUSE THE CODE IS PARTLY HARDWARE DEPENDENT . . . OUR DOC FILES ARE IN A WIERD LASER PRINTER MACRO LANGAUGES THAT WOULD LIKELY BE USELESS TO ANYBODY ELSE. (GA)

JEFF S. OK, THANKS . . . IT'S JUST THAT THE MANUAL UPDATES SEEM TO BE EXPENSIVE.

KARL WHEN WILL FORTRAN BE READY FOR THE COCO. OR SHOULD I SAY WHEN WILL THE COCO BE READY FOR FORTRAN?? (GA)

MICROWARE OUR NEW FORTRAN IS TOO BIG FOR LEVEL ONE SUCH AS COCO. SORRY, BUT CAN'T SPECULATE ON FUTURE TANDY PRODUCTS. (GA)

KARL OUT OF CURIOSITY, HOW MUCH MEMORY WILL IT TAKE TO USE THE NEW HITACHI GRAPHICS CHIP? (GA)

MICROWARE IT HAS PRIVATE VRAM - SIZE VARIES ACCOURDING TO THE # COLORS MAX VIRTUAL SCREEN SIZE, ETC. I THINK 128K IS A MINIMUM FOR NICE EFFECTS. (GA)

SUPER CONTROLLER

Features:

- * Gold contacts on all connectors.
- * Shielded metal box for low RF noise.
- * 4 28-pin sockets for software expandability.
- * Uses 2764 or 27128 EPROMS.
- * EPROMS are software selectable.
- * Internal Mini-Expansion Bus interface for:
 - Parallel Printer or
 - Real Time Clock Parallel Printer or
 - 80 Col Display Clock Parallel Printer or
 - EPROM Programmer or
 - User projects.
- * Complete Radio Shack compatibility.
- * New technology, no adjustments needed.
- * Very Accurate 16mhz High Speed Master Clock.
- * Needs 5 volts only, works on all COCOs or COCO IIs.

EXPANSION ADD-ONS:

PPRINT

The first is a Centronics Compatible Parallel Printer Adapter. This adapter will allow you to connect a Centronics compatible printer directly to your controller, leaving the serial port of your computer free for your modem.

RTIME

The second is a Real Time Clock. This is a clock chip that will keep the proper time, date, and year. A small battery keeps the time when the Computer is off, retrieve and set time by using simple Basic POKES. Also available with the Real Time Clock is the Centronics Compatible Parallel Printer adapter. Software to set the clock included.

MPROM

The third is a Mini EPROM Programmer. Yes, a low cost programmer that attaches to the disk controller. A must for the DISTO Super Controller. Program those often used utilities into EPROM and plug them directly into your controller. Will program 2764's or 27128's, a perfect mate for the DISTO Super Controller.

DISPLAY80

The fourth is a real knock-out. This is a three in one card. It's major function is to add an 80* 24 display to your computer. A feature packed package also includes RTIME and PPRINT. All in one neat package that fits inside the controller. Call for more information.

256K/512K SUPER RAM DISK

This is a ROM PAK the size of a typical controller. Inside this, low noise metal case, lives 256K/512K of memory and all the circuitry needed to access it as a RAM DISK. With proper software, this SUPER RAM DISK can be just like another disk drive. You can format it, save a file to it, load a file from it and delete files from it. In fact, anything that can be done on a regular drive, can be done on a RAM DISK, only faster. You see, being high-speed RAM, there is no hardware limitations on speed. It is much faster than even the fastest drive.

Another feature with the SUPER RAM DISK is that it has the same MEB as the Super Controller. That means that all of the add-ons that fit inside the controller will also fit into the RAM DISK. Note, a Multi-Pak is needed when using the RAM DISK with a disk controller.

OS-9 USERS

The OS-9 operating system is rapidly becoming a BEST SELLER. All the DISTO products are supported by OS-9 software. We have drivers for; PPRINT, RTIME, DISPLAY80, RAM DISK, and soon to come, HARD DRIVE. Just think of this, a floppy drive controller, a parallel printer port, the real time, an 80 column display, a 512K RAM Disk and a 20 megabyte Hard Drive, all in two slots of a multi-pak interface. AWESOME!



**SUPER
PRODUCTS**

CREDITS:

The DISTO Super Controller, add-ons and all its documentation are conceived and designed by TONY DISTEFANO. The DISTO Super Controller and add-ons are manufactured and distributed by;

CRC Inc.
10802 Lajeunesse Suite #102
Montreal, Quebec
Canada
H3L 2E8
1-514-383-5293

*DISTO and CRC Inc. are registered trademarks. The DISTO Super Controller and add-ons are copyrighted by DISTO. Radio Shack is a division of Tandy Corp. Color Computer is a trademark of Radio Shack. Dealer inquiries welcomed.

FOOD FOR BODY AND SOUL! OS-9 COMMUNITY BREAKFAST AT PALO ALTO

There's absolutely nothing that tops off a good meal like a good conversation. A drink won't do it, a good cigar won't do it, even a good movie won't do it. You could consider that the theme of the OS-9 Community Breakfast at the Palo Alto RAINBOWfest.

First there was the buffet breakfast itself, and the food was great. Then there was the conversation with the meal. Since the decision was made that there would be no head table, all of the UG officers were spread out, I was having fun with Lonnie Falk, Wayne Day and Dan Dornard at my table.

Brian Lantz got up to open the speakers forum, and the fun really began.

We started with some news about what's what in the UG, new developments to be expected, the announcement about the planning of a grant to develop software that could be purchased through the UG library, and developing revenue for additional software development.

Then the awards were made. A tongue in cheek look at what's going on in the OS-9 UG from its President. All of the awards were made for fun with the award recipients getting the most laughs.

Wayne Day was asked to make some comments on recent changes to the OS-9 SIG. Wayne announced his assumption of duties as the OS9 SIG's Sysop. Wayne contacted Jonathan Cluts and Pete Lyall during the Palo Alto RAINBOWfest to act as his assistant Sysops. Wayne says they are the real managers of the system. You can reach Wayne using his user id of 76703,376. Jonathan's id is 70235,457. I'll pass on Pete's when it's available.

I was asked to make a few comments on writing for the MOTD. If you haven't heard or read my pitch, read my column in the March issue of the MOTD, or join us in Chicago. The basics are that you should be a member of the UG, not just a reader of the MOTD. As Bill Turner says, "what you learn now you can share with others later on."

We finally got around to Paul Searby who was nothing short of fantastic! Paul kept his talk to the fundamentals, echoing much of what Dale Puckett talks about in his column this month.

Paul's thrust is a charge to the entire OS-9 community. We are the ones that are responsible for making OS-9 THE operating system of our time and beyond. Since every change to MS-DOS is an

inclusion of what's already in OS-9, and what OS-9 lacks compared to UNIX isn't a requirement of a multi-user/multi-tasking system, we should stop comparing OS-9 to UNIX. Add to that the fact that UNIX still hasn't been accepted throughout the market place, and Paul makes a very valid point.

Paul didn't stop with people like you and me. He went on to task the big guys like Microware and Tandy to make OS-9 a more accessible system for the new comer. People don't start programming in Assembly language. Computer companies have been marketing computers with BASIC for four or five years now. OS-9 comes with a good assembler and a handy editor, but neither is the same as a fundamental programming language. The suppliers of the operating system, according to Paul, have to give a programming language with the computer (I vote for Computerware's RANDOM Basic as an easy to learn language).

Paul's final charge was to make sure we (the UG) keep an open door for newcomers. Many of the people that join the UG are looking for answers. They don't know where to turn, because OS-9 is so far removed from the Color Computer. So what do you do? You keep it simple and make yourself more accessible. Through our attendance at RAINBOWfests, we're doing just that. We're responding to individual questions and making sure the answer is understood. We're making some of the best known names in the OS-9 community available to everyone around. We're just trying to respond to Paul's charge.

With all that, what could follow? Well, come on out to Chicago and meet the creator of OS-9, Ken Kaplan. Along with those of us at the OS-9 booth, you'll meet some of the real leaders of the OS-9 community. We'll have Brian Lantz, Dale Puckett, Steve Odneal, myself and a whole lot more.