

Announcing...

The 5th Annual Atlanta CoCofest

Holiday Inn, Northlake
October 1 & 2, 1994

Show Hours:

Saturday, Oct 1 - 9:00 am - 5:00 pm
Sunday, Oct. 2 - 9:00 am - 3:00 pm

Vendor Set-Up:

Friday, Sept. 30 - 6:00 pm - 9:00 pm
Saturday, Oct. 1 - 8:00 am - 8:45 pm

Admission: \$10.00 (Whole Show)

Reservations:

Holiday Inn, Northlake
(800) 465-4329 or (404) 938-1026

Sponsored by:

Atlanta Computer Society
P.O. Box 80694
Atlanta, GA 30366
BBS: (404) 636-2991

Volume 2, Issue 4

300

The "International" OS9 Underground[®] Magazine a FatCat Publication[®]

DEDICATED TO OS9/OSK USERS EVERYWHERE

IS OS-9 SUPPORT SINKING?



- BUILDING YOUR OWN C LIBRARIES
- CHECKS - CHECK REGISTER IN BO9
- HOW TO PUT TOGETHER AN INEXPENSIVE OSK MACHINE

**PLUS... SHOW YOUR SUPPORT OF OS9
(FREE BUMPER STICKER MAIL-IN OFFER INSIDE)**

Bob van der Poel Software

Great Stuff for OSK

Ved Text Editor - The most powerful editor for OSK. Latest version Includes an Integrated spelling checker and text formatter. Complete editing functions. Includes MVEF for changing Ved environment files. Works on any terminal **\$59.95**

Vprint Text Formatter - The ideal companion for Ved. This command base formatter supports proportional fonts, a powerful macro language, complete margin/indentation control and footnotes. **\$59.95**

Vmail - Written to manage our own mailing lists, this mailing list program is slick and fast. Using a proprietary system of linked lists and external indexes, any name can be found and printed instantly. **\$49.95**

Cribbage - The classic card game! Pit your skill against the computer. Requires KWindows and mouse. **\$19.95**

Magazine x System - Find those magazine articles...fast. File compatible with our level II product. Comes with 300K of Coco files. **\$19.95**

All prices shown are for single user licenses. Please write for multi-user pricing or a free catalog (level II or OSK). Please add \$3.00 shipping and handling to all orders (\$5.00 overseas). To order, please send your check or money order and preferred disk format to:

Bob van der Poel Software

P.O. Box 57
Wynndel, BC
Canada V0B 2N0
Telephone

P.O. Box 355
Porthill, ID.
USA 83853
604-866-5772



SUB-ETHA SOFTWARE

Come see us at the 5th Annual Atlanta CoCoFest!

Invaders09 by Allen Huffman *NEW* An arcade classic "blast" from the past. Invaders from space! Blast waves of enemies with only your shields as protection in this new high-speed OS-9 game. Send \$10 extra for source code. CC3, OS9L2 (Available Oct. 94) **\$19.95**

MultiBoot V1.04 by Terry Todd & Allen Huffman *NEW VERSION!* Have up to 16 OS9Bootfiles on one disk! PowerBoost & NitroS9 compatible. CC3, OS9L2 **\$19.95**

Towel V1.01 by Allen Huffman *NEW VERSION!* Keyboard/Mouse driven file manager. All commands/colors configurable. User-definable menu for custom options. Runs on high-speed text screen under the EthaWin interface (Included). CC3, OS9L2, Mouse Opt. **\$24.95**

1992 CoCoFest SIMULATOR V1.02 by Allen Huffman *NEW VERSION!* Take a walking tour of the '92 Atlanta CoCoFest with this graphics adventure. Dozens of digitized images. Runs on a 640x192 graphics screen. Now uses compressed graphics and has scoring. CC3, OS9L2, 500K+ Disk Space \$ 9.95 MM/1 (Old Version) **\$14.95**

Copy Cat from StrongWare - Simon says "match colors/tones in order." MM/1 or 100% K-Windows Compatible **\$14.95**

SUB-ETHA SOFTWARE

Add #250 S&H P.O. Box 152442 • Texans add 8.25% Lufkin, TX 75915
Write us for more info! E-Mail: coco_sysop@genie.geis.com

Wittman Computer Products

HARDWARE • SOFTWARE • CONSULTING

Full MM/1 and OS-9/68xxx support

Software

Custom Point-Of-Sale and accounting support

Bob van der Poel Software

GCal - a Graphical Calendar

DeskTamer v2.0 - a premier personal information manager
(calculator, filer, notetaker, full graphical interface)

LaTerm.LaDial -terminal package

Hardware

68340 Accelerator Boards

I/O Paddle Boards / Midi Boards

Floppy and Hard Drives

Cases & Power Supplies

Data / Fax Modems

Keyboards, Memory & Mice

Point-Of-Sale Equipment

We look forward to seeing you at the
5th Annual CoCoFest in Atlanta

William L. Wittman, Jr.

Wittman Computer Products

873 Johnson Road

Churchville, NY 14428

(716) 494-1506

our E-Mail addresses are:

...!swamps.roc.ny.us!mm1wny!wlwj
ww2150@acsprl.acs.brockport.edu

a terminal based system. Well, to my surprise I found there was more than enough software to keep me happy.

This system comes with a number of powerful utilities (similar to the OS9 6809 but enhanced) including a good text editor, Umacs. There are also a variety of good word processors available including Ved which I quickly purchased because I use it on my COCO. My next quest was to get a terminal program. Since I was running low on cash I downloaded Stern which works well, but offers limited capabilities. There are however other commercial programs, like Info Express, available if more powerful terminal software is needed. Having achieved my basic needs, I decided to install SCREEN which would allow me to run several full screen windows and downloaded several programs including SC - a spreadsheet program. What is most important to note is that software is available.

While it is true that the software availability is not the same as the the IBM world, nevertheless, there is good software available. The bottom line is, I love OSK, it is a challenging and fun environment to work with. OSK offers new and exiting possibilities for users who want to move beyond the COCO or who just want a faster OS9 machine.

-JAMES POTTAGE
97 Leopold St.
Trenton On
Canada



About the author:

James Pottage has been a Coco OS9 user since 1987 and has just recently begun programming in C. James is 31 and works for the Department of National Defense as an Air Traffic Controller. He used the COCO as a word processor at his university and is presently using the PT68K4 for word processing and programming.

COMING NEXT MONTH: CocoFest Picture Issue

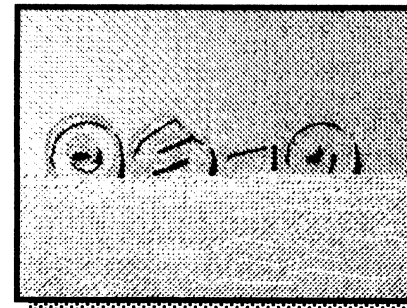
Advertiser's Index

Vendor	Page
Bob van der Poel Software	IFC
Sub-Etha Software	IFC
New Discoveries Request	5
OS9 Underground Advertising	11
Fat Cat Publications	16
Wittman Computer Products	IBC
Atlanta Computer Society	BC

The "International" OS9 Underground Magazine

A Fat Cat Publication®

Volume 2, Issue 4



Is Support for OS9 Sinking?

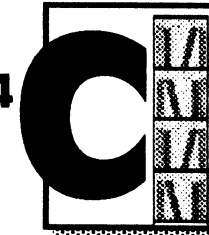
Comentary on OS9 Support.
by Alan Sheltra

PAGE 4

Building a C Library - Part 4

by Bob van der Poel

PAGE 6



Checks Part 2 by Wayne Campbell

PAGE 12



Building a Complete OSK Machine for Under \$800 by James Pottage

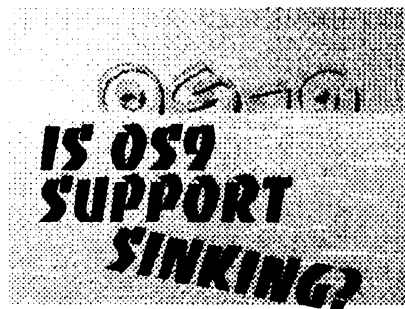
PAGE 12

New Discoveries Ad Index

Page 5
Page 18

Staff Editor: Alan Sheltra
Asst Editor: Steve Secord and Jim Vestal
Staff Writers: Bob van der Poel, Scott McGee,
Boisy Pitre, Allen Huffman, James Pottage

Fat Cat Publication and the "International" OS9 Underground Magazine and it's logotypes are registered trademarks. Subscription rates are \$18.00/year for 12 issues (\$22 Canadian, \$27 overseas) Fat Cat Publications is located at 4650 Cahuenga Blvd, Ste #7, Toluca Lake, CA 91602 (818) 761-4135 (818) 769-1938 (fax). The contents of these pages are copyrighted. Photocopies or illegal reproduction of this magazine in part or whole is stricky prohibited without prior written permission.



Commentary by Alan Sheltra

Support is a two-way street. You need the vendor's and software/hardware developers to create and maintain all those neat utilities and applications for you, and they need you... to purchase, those products... and tell your friends about them.

Magazines and newsletters are one of the conduits by which both parties can reach each other. Vendors/developers through ads and you by sending in your comments, kudos and criticism.

Other conduits are the Electronic services and networks, which by their immediate nature, make them ideal for almost real time news and feedback.

Not everyone is on these services or nets, so that leaves the Magazines and Newsletters as their only source of news and at last count, that only makes 2 magazines left covering OS-9, "World of 68xxx" and the Underground. ("Metamorphosis", published by Mark Griffith, folded in August).

There are of course many specialized newsletters around the country, the biggest being the MOTD, the OS-9 User Group's newsletter, CoCo123 published by Glenside and the OS-9 Newsletter produced by the PNW Group. (I'm sure there are other smaller newsletters that I am not aware of or have overlooked).

Subscribe to all the mags and newsletters you can. They're your all IMPORTANT line of communication.

Mark Griffith, as you may have heard has completely pulled out of OS9 and while I don't agree with his method of "leaving" I do see some of his reasons for leaving. Here's an excerpt from an article he sent me (not really an article, but more a good-bye letter).

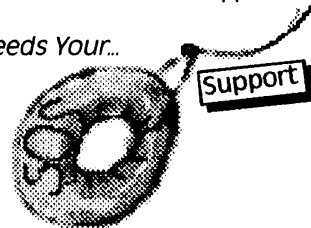
"As of this writing, I am officially out of the OS-9 computer hardware and software business. The "Dirt Cheap Computer Stuff Company" is being shut down, and the "Metamorphosis" magazine is out of print. I've taken this step with many misgivings, but no regrets..."

Mark has made many contributions to OS-9, and I will be sorry to see him leave our community.

Basically, Mark (and many others) as a software vendor was discouraged by dwindling software sales and lack of any real progress in development of OS-9. I can sympathize with him on the first part, AniMajik, my software company barely made enough to recoup it's advertising costs this year. But I have always been involved with OS-9 because it was a *hobby* and an *enjoyment* to me. Realistically, the market is much too small for Mark's financial expectations.

Don't let this discourage you though (especially you software/hardware vendors), it just means if we want OS-9 to continue to work, we have to first love it (I'm sure most of you do), and work a little harder to support it!

OS-9 Needs Your...



Building a Complete OSK System for Under \$800

by James Pottage

There has been a lot of discussion in the past about upgrading to OSK. Well, for what its worth, here is my two cents - IT IS POSSIBLE TO BUY A GOOD SYSTEM ON A LIMITED BUDGET. Further, there is ample software available that will meet the needs of most users. Last summer I began looking at upgrading to OSK.

After looking at the systems from Frank Hogg's, Delmar and Peripheral Technology, I decided it was time to get my feet wet.

Unfortunately I had a limited budget, therefore, the cost of the system became my biggest consideration. After consulting with Fred from Peripheral Technology, I believed I had found a system that I could both afford and enjoy. Peripheral Technology offers two low budget mother boards that will run OS9 68000. Both are based on the motorola 68000 processor. The first board is the PT68 K2 board which has a maximum of one meg memory, four serial ports, a parallel port and 8 IBM slots. It runs at 8 megahertz and will support, among other things, an IDE drive controller card, an MFM controller card, and a VGA card. Each of these cards are specific IBM controllers, and their model numbers can be obtained from Peripheral Technology, or you can order directly from them. Further, additional serial cards can be purchased from Delmar. The other board is the PT68 K4 board. This system has all the same features as the K2, plus it will support up to four meg of Ram and runs at 16 megahertz. Both systems are well designed and offer a solid base from which to build an OSK machine.

After looking at the prices, \$125 for a K2 vs \$249 for a K4, I decided I could afford the K4. Next I needed a Hard drive. I opted to use one of my two MFM drives connected to my COCO and ordered an MFM drive card for \$59 from Peripheral Technology. The next step was to get a case, power supply and disk drive. I found a high density 3.5 inch drive from a local store for \$59, and a case and power supply for \$89. I already had an old cybernex xl87m terminal which I decided to use as my monitor and keyboard. Finally, I needed an RS232 connector and Parallel Port cable which cost \$30.

So lets recap the prices:

PT68K4	- \$249.00
Power supply & case	- \$89.00
Terminal	- \$0.00
Hard Drive	- \$0.00
Hard Drive Controller	- \$59.00
3.5 Floppy	- \$59.00
Cables	- \$30.00
OS9 6800	- 299.00
Total	- \$755.00

The total cost was under \$800.00, *Amazing!* In mid February I had found the money and by the beginning of March I was up and running with an OSK machine.

Now as you have no doubt noticed the system I have put together does not support graphics. This concerned me, because I wondered how much software there was that would support

allows a process writing through a SCF /dev device to not block even if another process is waiting for input. This is useful for logging important messages to other terminals which may be in read mode via a shell.

The DDF manual is comprehensive in technical content, but lacks a specific set of instructions on how to install DDF. I did find Ark Systems' Ikuko

Kato very helpful when I needed assistance with some installation problems that occurred. All things considered, DDF is a useful product for those who need its functionality.


-Boisy G. Pitre
Internet: boisy@os9er.waukee.ia.us
Delphi: BOISY



SHOW 'EM YA LOVE OS-9!

Now available from Fat Cat Publications,
Bumper stickers to show your support!
(and a way to get one FREE, see below)

Styles available:

- (A) "I Support OS-9"
- (B) "OS-9: Kickin' Butt in Realtime"
- (C) "I  OS-9"
- (D) "OS-9ers do it in Realtime"
- (E) "OS-9ers don't do Windows"

(All Bumper Stickers are black print on Bright Orange peel 'n stick material)

Bumper stickers are 75¢ each (3 for \$2.00).

Please include 29¢ S&H per order (40¢ Can, 98¢ Overseas)

FREE OFFER

Fill out COMPLETELY and send in the OS-9 Survey Form (included with this issue) and include either a Self-Addressed, Stamped Envelope or a 29¢ stamp (40¢ Can, 98¢ overseas) and we'll send you an "I Support OS-9" Bumper Sticker Free!

Mail Survey and your SASE to: Fat Cat Publications
4650 Cahuenga Blvd., Ste #7
Toluca Lake, CA 91602

NEW DISCOVERIES

512K SIMM MEMORY UPGRADE - 512K \$44.95 ZERO K \$39.95

NOW AVAILABLE FROM NORTHERN XPOSURE. MOST OF THE PROFIT GOES TO THE TORONTO CLUB.

RUNS COOLER AND USES LESS POWER THAN STANDARD UPGRADES, USES TWO 8 OR 9 CHIP 256K SIMMS, 80-150NS

CONTACT: NORTHERN XPOSURE (613)736-0329 7 GREENBORO CRES, OTTAWA, ON, CANADA K1T 1W6

VINTAGE COCO/OS-9 SOFTWARE STILL AVAILABLE

12 CoCoPro! OS-9 PROGRAMS ARE AVAILABLE IN A PACKAGE DEAL FOR

ONLY \$55 + \$5 S/H. ALL DOCS COME WITH THE PROGRAMS OF COURSE...DATA WINDOWS, NEWSPAPER09, MULTI-MENU AND A HOST OF OTHERS ARE IN THE PACKAGE.

CONTACT: RICK'S COMPUTER ENTERPRISE - P.O. BOX 276 LIBERTY, KY 42539

COCO SOFTWARE STILL AVAILABLE FROM TANDY

MOST TANDY COCO PROGRAMS ARE STILL AVAILABLE THROUGH A LITTLE KNOWN SPECIAL ORDER CONSUMER MAIL CATALOG. THERE IS A BOOK WHICH MOST RADIO SHACK STORES HAVE HIDDEN BEHIND THE COUNTER FOR ORDERING THIS TYPE OF "OLD" MERCHANDISE. THE PRICES HAVE BECOME QUITE REASONABLE; CALL (800) 321-3133 AND ASK THEM TO CONNECT YOU TO THE SPECIAL ORDER LINE FOR COCO SOFTWARE.

NEW FREE MM/1 SOFTWARE

SUB-ETHA SOFTWARE AND BLACKHAWK ENTERPRISES ARE PLEASED TO ANNOUNCE ADDITIONAL FREE SOFTWARE BEING INCLUDED WITH ALL NEW MM/1 COMPUTER SYSTEMS. SUB-ETHA'S FLEXIBLE CHECKBOOK* PROGRAM WILL NOW BE INCLUDED WITH ALL NEW MM/1 COMPUTER SYSTEMS. JOEL MATHEW HEGBERG HAS COMPILED A COLLECTION OF HIS MOST POPULAR PUBLIC DOMAIN SOFTWARE INTO A NEW DISK CALLED "JOEL MATHEW HEGBERG'S MM/1 P.D. ANTHOLOGY DISKETTE," ALSO INCLUDED. SUB-ETHA SOFTWARE'S IMMENSELY POPULAR MINIBANNERS. EXISTING MM/1 OWNERS WHO WISH TO OBTAIN JOEL'S P.D. ANTHOLOGY MAY OBTAIN IT THROUGH BLACKHAWK ENTERPRISES FOR A \$12.50 MEDIA/COPYING/SHIPPING FEE. EXISTING MM/1 OWNERS WHO WISH TO OBTAIN CHECKBOOK* OR MINIBANNERS FOR K-WINDOWS MAY OBTAIN IT THROUGH SUB-ETHA SOFTWARE FOR \$12.50 (\$10.00 MEDIA/COPYING + \$2.50 S&H).

BLACKHAWK ENTERPRISES - P.O. Box 10552 - ENID, OK. 73706 - (405) 234-2347

SUB-ETHA SOFTWARE (OSK/MIDWEST DIVISION) - 936 N. 12TH ST. DEKALB, IL. 60115-2516
(815) 748-6638

VENDORS AND SOFTWARE AUTHORS... (COMMERCIAL AND SHAREWARE)

Got some new wares we need to hear about? Perhaps a new shareware program you want everyone to know about? Please send information about them to New Discoveries to:

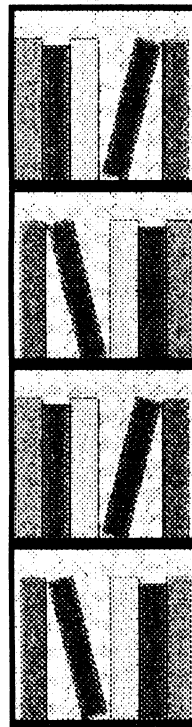
Fat Cat Publications
New Discoveries
4650 Cahuenga Blvd., Ste #7
Toluca Lake, CA 91602

Email: discover@zog.wa.com

Fax: (818) 769-1938

Creating Your Own C Library

by Bob van der Poel



Welcome to our final installment dealing with C libraries. In earlier parts we discussed the purpose of a library, how to create your own libraries, and showed some same modules (mostly in 68000 assembler). This month we'll give you a few more modules...and digress into some personal library philosophy.

First, some useful functions:

One nuisance I run into frequently is reading lines in from a file. Since most files have lines ending in carriage returns, and the simplest way to store strings in C programs is to have them terminated in a 0...

In Review

DDF - /dev

Device File Manager for OS9/680X0

The gap between OS-9 and UNIX has just become narrower with a new product from Ark Systems USA. They have introduced DDF, a package which allows your OS-9/68K system to access devices in the same way that UNIX systems do. DDF does not replace other file managers, but rather coexists with them, giving you flexibility to use both OS-9 specific and UNIX-style device names interchangeably.

The UNIX equivalent of an OS-9 device descriptor is known as a device file. These files are found in the /dev directory of a UNIX disk. All devices accessed by UNIX are referenced in this directory. For example, writing to a window device from a shell under an OS-9 machine may be done this way:

```
echo test >/w5
```

...whereas on a UNIX system, the window device is accessed as follows:

```
echo test >/dev/w5
```

To further expand upon the UNIX file structure, DDF allows device 'aliasing' by employing a data module strategy. The 'addf' utility, which is included in the package, facilitates the building of this data module from a text file, /dd/SYS/devalias, containing user-defined aliases.

For instance, you can refer to your terminal ports in standard UNIX form as /dev/tty0 and /dev/tty1 (instead of /dev/t0 and /dev/t1). Some sample entries are provided below.

```
# RBF devices
fd1200.0:pc0
fd1440.1:pc1
fd360.0:pd0
fd720.1:pd1
```

```
rd0:h0
rd1:h1
```

```
# SCF devices
tty0:t0
tty1:t1
```

DDF consists of the 'ddf' file manager and 'dev' device descriptor. 'dev.dev' is the optional alias data module which DDF refers to when resolving device aliases. All of these modules can be instantly used when loaded into memory. This also lends DDF to be used quite conveniently in ROMable applications.

Using the Dir command, you can receive a listing of devices which are currently under DDF's control. dir /dev returns all devices. Wildcards are accepted, and an extended directory listing gives driver specific information, such as static storage size and device table entry address.

Besides UNIX consistency, DDF also provides some very interesting features:

- /dev/tty always accesses the current terminal device, no matter what devices the standard input and standard output paths are redirected to.
- /dev/console retrieves the system console device name from the INIT module
- A corrupted SCF terminal path can be fixed from another terminal using TMODE. If you encounter a program which prematurely aborts, leaving the path in an unstable state, you can go to another window or terminal, and reset the defective path by: tmode normal/dev/terminal_device.
- DDF has "interruptive writing" which

```

1159     IF NOT(entry.check) THEN      1330     (* Toggle Tax Deductable
1166     PRINT                          Flag
1168     PRINT "Is this a                134D     IF SUBSTR(key,"Tt")=0 THEN
deposit? ";                            135E     PRINT
1180     REPEAT                          1360     IF entry.deductable THEN
1182     (* REPEAT                        136C     entry.deductable:=FALSE
118B     (* count=INKEY(#0)              1376     ELSE
119E     (* UNTIL count=0                137A     entry.deductable:=TRUE
11AE     (* GET #0,char                  1384     ENDIF
11BC     RUN Inkey(char)                 1386     ENDIF
11C6     UNTIL SUBSTR(char,"           1388     (*
YyNn"+CHR$(13))=1                      138B     EXITIF SUBSTR(key,"Qq")=0
11DD     IF SUBSTR(char,"Yy")=0         THEN
11EE     THEN                            139C     key:=""
11F8     entry.deposit:=TRUE            13A4     ENDEXIT
11FC     ELSE                            13A8     ENDOLOOP
1206     entry.deposit:=FALSE           13AC     (*
1208     ENDIF                            13AF     SEEK #file,rec_num*SIZE
120A     new:=entry                      (entry)+SIZE(control)
1212     (*                               13C5     PUT #file,entry
1215     (* Revise Balances              13CF     GOSUB 110
1227     GOSUB 70                          13D3     ENDIF
122B     ENDIF                            13D5     (*
122D     (*                               13D8     (* Remove (Delete) a Record
1230     (* Change Description            IF SUBSTR(key,"Rr")=0 AND
1245     IF SUBSTR(key,"Rr")=0 THEN      control.num_recs=0 THEN
1256     PRINT                            GOSUB 110
1258     INPUT "New Description: "       1412     PRINT
,entry.desc                             1414     PRINT "Remove Entry"
1275     temp_desc:=""                   1424     PRINT
127C     FOR counter:=1 TO LEN           1426     (*
(entry.desc)                             1429 30   INPUT "Record # to
char:=MID$(entry.desc                   Remove: ",temp_rec
,counter,1)                              1448     IF temp_rec="" THEN
12A2     IF char="a" AND char="         1454     rec_num:=screen_pos
z" THEN                                  145C     ELSE
12B7     char:=CHR$(ASC(char)           1460     rec_num:=VAL(temp_rec)-1
-32)                                     146E     ENDIF
12C4     ENDIF                            1470     IF rec_num=0 OR rec_num
12C6     temp_desc:=temp_desc           control.num_recs THEN 30
+char                                     148A     FOR
12D2     NEXT counter                    count:=control.num_recs-1
12DD     IF temp_desc="VOID"            TO rec_num+1 STEP -1
THEN                                       14AB     SEEK #file,count*SIZE
12ED     entry.desc:=temp_desc          (entry)+SIZE(control)
12F9     entry.amount:=.0               14C1     GET #file,entry
1308     new:=entry                      14CB     SEEK #file,(count-1)*SIZE
1310     (*                               (entry)+SIZE(control)
1313     (* Revise Balances
1325     GOSUB 70
1329     ENDIF
132B     ENDIF
132D     (*

```

Part 3 Continued
Next Month...



```
/* creadln.c */
```

```
/* This is exactly the same as the OS9 readln(), however a null
is appended to the end of the string. If the string read ends
in a carriage return it is replaced with a 0.
```

```
Returns -1 on error
0 on EOF
else, number of bytes read
```

```
*/
```

```
creadln(path, buffer, t)
int path;
register char *buffer;
register int t;
{
    if(t=readln(path, buffer, t)>0) /* read a line, if no error */
    {
        /* point buf pointer one */
        buffer++; /* past end of read data */
        /* if last char was CR backup one */
        if*(buffer-1)=='\n' buffer--;
    }
    *buffer='\0'; /* convert CR to NULL */
    return t; /* or append NULL past data */
}
```

Of course, this only works if you are using low-level, unbuffered I/O. If you are using buffered, C I/O then this might be more useful:

```
/* fgets_nlc.c */
```

```
/* fgets() line and strip trailing CR
```

NOTE: if line read does not end in a CR then we read until a CR is found. This truncates the input line...

```
*/
```

```
#include <stdio.h>
#include <strings.h>
```

```
char *
fgets_nlc(s, n, file)
char *s;
int n;
FILE *file;
```

```

{
register char *p, *ret;
register int c;

ret=fgets(s, n, file);
ifret && (p=index(s,'\n')) *p='\0';
return ret;
}

```

If you want to write strings containing carriage returns using low-level I/O, you'll find this function handy (no, I didn't include it just to get mail about using goto!):

```

/* cwrite.c */

/* Write a NULL terminated string to the specified path.
Multiple carriage returns are supported by multiple calls to writeln()
*/

cwriteln(path, s)
int path;
register char *s;
{
register int t;
register int total=0;
register char *p;

while(*s)
{
p=s;
while(*p && *p++!='\n'); /* advance to EOL or CR */
t=writeln(path, s, p-s); /* print from last pos to here */
if(t--<-1) /* error check! */
{
total=t;
goto EXIT;
}
total+=t; /* track # printed */
s=p; /* start next segment */
}
EXIT: return total;
}

```

```

0DE5 GOSUB 110
0DE9 PRINT
0DEB PRINT "Edit Entry"
0DF9 PRINT
0DFB (*
0DFE 20 INPUT "Record # to Edit "
temp_rec
0E1B IF temp_rec="" THEN
0E27 rec_num:=screen_pos
0E2F ELSE
0E33 rec_num:=VAL(temp_rec)-1
0E41 ENDIF
0E43 IF rec_num<0 OR rec_num
>control.num_recs THEN 20
0E5D SEEK #file,rec_num*SIZE
(entry)+SIZE(control)
0E73 GET #file,entry
0E7D (*
0E80 LOOP
0E82 GOSUB 110
0E86 PRINT
0E88 PRINT " ";
0E8E IF entry.check THEN
0E9A PRINT USING "s5"
entry.number;
0EAA ENDIF
0EAC PRINT TAB(8); entry.date;
0EB9 PRINT TAB(17); entry.desc;
0EC6 IF entry.clear THEN
0ED2 PRINT TAB(46); "***";
0EDC ENDIF
0EDE IF entry.deductable THEN
0EEA PRINT TAB(48); "***";
0EF4 ENDIF
0EF6 IF entry.amount<0 THEN
0F09 IF NOT(entry.deposit)
THEN
0F16 PRINT TAB(50);
0F1C PRINT USING "r9.2"
entry.amount;
0F2E ELSE
0F32 PRINT TAB(60);
0F38 PRINT USING "r9.2"
entry.amount;
0F4A ENDIF
0F4C ENDIF
0F4E PRINT TAB(70);
0F54 PRINT USING "r10.2"
entry.balance

0F66 (*
0F69 (* Input Routine for Edit
Options *)
0F8A PRINT
0F8C PRINT "INumber IDate
Description ITax
Deductable IAmount
IQuit -> ";
0FDD REPEAT
0FDF (* REPEAT
0FE8 (* count:=INKEY(#0)
0FFB (* UNTIL count<0
100B (* GET #0,key
1018 RUN Inkey(key)
1022 UNTIL SUBSTR(key,"
AaDdNnQqRrTt")=1
103D PRINT
103F (*
1042 (* Change Amount
1052 IF SUBSTR(key,"Aa")<0 THEN
1063 PRINT
1065 INPUT "New Amount: "
entry.amount
107D er:=ERR
1083 new:=entry
108B (*
108E (* Revise Balances
10A0 GOSUB 70
10A4 ENDIF
10A6 (*
10A9 (* Change Date
10B7 IF SUBSTR(key,"Dd")<0 THEN
10C8 PRINT
10CA INPUT "New Date: "
entry.date
10E0 ENDIF
10E2 (*
10E5 (* Change Check Number
10FB IF SUBSTR(key,"Nn")<0 THEN
110C PRINT
110E INPUT "New Number: "
entry.number
1126 IF entry.number="" THEN
1135 entry.check:=FALSE
113F ELSE
1143 entry.check:=TRUE
114D entry.deposit:=FALSE
1157 ENDIF

```




Checks

Part II

by Wayne Campbell

In part two, we will cover the functions Add, Edit, Insert, and Remove. Before getting into the specifics of each function, I need to state that the functions described here use GOSUB subroutines that are not included in this installment of the source code. Those subroutines will be covered in part four. Insert this segment of code where the first break in the code of part one OCCURS. *Editor's Note: SRC will be made available from the listserv@zog.wa.com, see page 18 for details.*

Checks SRC Part II

```

OC62  (*
OC65  (* Add Records
OC73  IF SUBSTR(key,"Aa")>0 THEN
OC84  LOOP
OC86  GOSUB 110
OC8A  PRINT
OC8C  PRINT "Add Entry"
OC99  PRINT
OC9B  (*
OC9E  INPUT "Check Number (Enter
      Q to quit): ",entry.number
OCCA  EXITIF entry.number="" AND
      SUBSTR(entry.number,"Qq")>0
      THEN
OCE8  ENDEXIT
OCEC  (*
OCEF  (* Record Data Input Routine
OD0B  GOSUB 90
OD0F  (*
OD12  SEEK #file,control num recs
      *SIZE(entry)+SIZE(control)

```

```

OD2B  PUT #file,entry
OD35  control.num_recs:=
      control.num_recs+1
OD47  SEEK #file,0
OD50  PUT #file,control
OD5A  end_rec:=end_rec+1
OD65  IF
end_rec=control.num_recs
      THEN
OD75  end_rec:=end_rec-1
OD80  ENDIF
OD82  IF end_rec>14 THEN
OD8E  start_rec:=end_rec-13
OD99  ENDIF
OD9B  screen_pos:=end_rec
ODA3  ENDOLOOP
ODA7  (*
ODAA  key:=""
ODB2  GOSUB 110
ODB6  ENDIF
ODB8  (*
ODBB  (* Edit Records
ODCA  IF SUBSTR(key,"Ee")>0 AND
      control.num_recs>0 THEN

```

Just looking at this code again, I realize that the goto could be replaced by a break. But I think I'll leave it as is—it appears more clear to me as is. Of course, you don't need the equivalent for buffered I/O since

printf() does it quite nicely. Random number generators are probably more written about than any other topic in C. Just so that I don't get left out, here is one I find quite reliable and useful.

```

/* rnd.c */
#include <time.h>
rnd(x)
int x; /* max value to return */
{
    register long temp;
    static union
    {
        struct sgtbuf tbuf;
        long lnum12;
        int flag;
    }seed;
    if(!seed.flag) gettime(&seed.tbuf); /* seed the pattern */
    seed.lnum101+=0x1234567; /* set next pattern */
    seed.lnum11+=0x89abcde;
    temp=seed.lnum101+seed.lnum11; /* get number */
    if(temp<0) temp=-temp; /* make sure >=0 */
    return (int)(temp % x); /* return the number */
}

```

Note that this function works much like the RND function available in BASIC. You pass it a maximum value and it returns a number between 1 and that value.

I present the following function with philosophy attached...I'm not sure if it should be a library function, or if it is easier to rewrite and include in your application:

```

/* strsave.c */
/* This will stuff a string into malloc'ed memory. */
#include <malloc.h>
char *
strsave(s)
char *s;

```

```

{
    register char *p;
    p=malloc(strlen(s)+1); /* get memory */
    if(p) strcpy(p,s);    /* if success, stuff string */
    return p;
}

```

The problem? Well, quite often you might want the function to just terminate the program if you run out of memory. In this case, you could add the following line after the strcpy():

```
else exit(-1);
```

Or, you might want to call your own error/terminate routine. If you use the function as is, you do have to check each call to make sure a valid pointer is returned (just like you do with malloc() calls).

So, this is where the sixty thousand dollar question comes! Should you compile your own functions into a library, or just link them into your application's source files?

On the library side are two obvious pluses: It is quicker to have the code in a library and you are more likely to have debugged the function before adding it to the library (maybe). On the other hand, if you end up making minor modifications all the time, then maybe you'd be better off just having the code in a source "library".

What I have elected to do is to move most of my useful, reuseable functions to libraries. This forces me to write tight, debugged code. Then, if I want to use a slightly different version, I grab the source, make the changes and add it to the application source files. One thing I SHOULD do, but don't (and I know it will cause me grief in the future) is to use a different name for the modified function. For example, if I change strsave() to call exit() I should rename it to strsave_exit() or something equally creative. That way I (and other readers of my code) should note that this is not the "standard" Bob van der Poel strsave().

Looks like we have room for a bit more code. So, here are two alternate string comparison functions. The first, does its compare without regard to case. Hence, "BOB" will equal "Bob".

```

/* strcmp.c */
/* Compare two null-terminated strings ignoring case */
#include <ctype.h>
strcmp(s1, s2)
register char *s1, *s2;
{
    register int c1, c2;
    for(;;)

```

```

{
    c1=toupper(*s1++);
    if(c2=toupper(*s2++) || !c1 || c1!=c2) break;
}
return c1-c2;
}

```

And this one will do the same compare, but you get to add in the number of characters to compare:

```

/* strcmp.c */
/* Compare two null-terminated strings ignoring case */
/* Stop compare after count characters. */
#include <ctype.h>
strncmp(s1, s2, count)
register char *s1, *s2;
register int count;
{
    register int c1, c2;
    while(1)
    {
        if(count--<=0) return 0;
        c1=toupper(*s1++);
        if(c2=toupper(*s2++) || !c1 || c1!=c2) break;
    }
    return c1-c2;
}

```

(EOF)

If you have been following along in this series (and have a 68K compiler) you can change the toupper() calls in these functions to f_toupper(). You can take out the #include <ctype.h> in this case, too.

Well, that's it for another issue. Next time? Well, right now I have no idea.

But reader suggestions are encouraged (especially at times like this when the idea well is on the dry side). My addresses are PO Box 355, Porthill, ID 83853 or PO Box 57, Wynndel, BC, Canada V0B 2N0 or, electronically, at Compuserve 76510,2203 or on the INTERNET at bvd@zog.wa.com.



You Need to Advertise!

New LOWER Rates

1 Time Full Page Ad Only \$10.00! 3 months for \$20.00
for details Call (818) 761-4135 • (818) 769-1938 (Fax)