

Published Bi-Monthly

Vol. 2 #2 Nov./Dec. 1988
\$2.75 U.S.A.
\$3.75 Canada

CoCo Clipboard Magazine

THE NEWEST, MOST INDEPTH MAGAZINE FOR TANDY'S COLOR COMPUTER 2 & 3

GOOD TIDINGS

OF GREAT

JOY

Multi-View Icon Editor

Reader Mail

Using OS9 Files

Clip-Ettes

The Assembly Line

Product Reviews

CoCo and The Model 100



J Bennett '88



◀◀ GIMMESOFT ▶▶



A new generation of Color Computer products

V-Term Terminal Emulator (128k or 512k CoCo III only) ◀◀ IMPROVED!

V-Term is one of the most advanced terminal programs for the CoCo III ever!!!
FEATURES: VT-100, VT-52, Vidtex(includes RLE graphics display), and standard CRT emulations. Full use of 512K, 80X28 text or graphics characters, Windows & Multi-tasking(Disk Basic), RAMDISK like buffer, Xmodem, Xmodem-CRC, Ymodem, Xon/Xoff, Monochrome monitor support, Capture buffer, Snapshot, Conference mode, Serial/RS-232/DCModem Pak, 35/40/80 Tracks, Over 56 pages of docs! Disk \$39.95 Ver 02.00.00 upgrade \$6.95



MAX-10

(CoCo III only) The dazzling Word Processor and document creator for the CoCo III! MAX-10 is the perfect partner for CoCo MAX III! Mix graphics and text to get great looking newsletters, flyers, term papers, ect... MAX-10 \$79.95 CoCo Max III \$74.95 or buy BOTH for \$139.95 and save!

MAXSOUND

(128k or 512k CoCo III only)

See June '88 review

Turn your CoCo III into a REAL digital audio sampler with HIGH quality audio reproduction. Easily add exotic effects, ECHO, stuttering, speed shifting, sequencing, and reverse audio to BASIC or ML programs or GRAPHICS! Now includes Data Compression. Imagine recording any Voice, Music, or Sound effect and being able to use these DIGITAL recordings in your own programs! 3 disk sides includes: INTERFACT/BIN - ML driver for sound effects. G&M/BAS - Adds sound effects to Graphics. SHOWTIME and DEMO disks. SCOPE/BAS - Turns CRT into a Digital Oscilloscope to look at MAXSOUND waveforms. CALL TO HEAR 'OVER THE PHONE' DEMO \$59.95

MAXSOUND SOUNDTRACKS W/GRAPHICS (MAXSOUND program NOT req'd) Call for Titles \$5.95

GRAPHICS-25

(512k CoCo III only)

Great with MAXSOUND and/or CoCo Max III!

Utilize the FULL 512k memory range of your CoCo III from BASIC for graphics! Create up to 25 ONBOARD HIRES SCREENS! Six new BASIC commands allow instant display switching while secretly drawing other screens. Save and load screens to and from disk. Copy one screen to another. Fast Graphics action, Smooth animation, and 100% Machine Language code. Requires DECB 1.0, DECB 1.1, or FKEYS III. Complete with documentation. Disk .. \$24.95

MULTI-LABEL III

(CoCo III only)

See July '87 review. An easy to use, versatile label creating program including many new CoCo III features. Print multiple fonts on each label! Disk \$16.95

FKEYS III

(CoCo I/II/III)

See April '87 review. A user friendly, user programmable function key utility that creates up to 20 function keys. Includes an EDITOR, DOS mods, DISABLE, and it's EPROMable! Disk \$19.95

SIXDRIVE

(CoCo I/II/III)

This machine language utility modifies DECB 1.0, 1.1, FKEYS III, or ADOS to allow the use of 3 double-sided drives (or 2 D/S drives and J&R's RAMDISKS) as 6 S/S drives. Disk .. \$16.95

AUTO DIM

(CoCo III only)

See Jan. '88 review. This hardware device protects your monitor, or TV from IMAGE BURN after a few minutes of inactivity from your keyboard. Illustrated and easy to install. Hdwe \$29.95

MPI-CoCo Locking Plate

(CoCo III only) Now 2 styles

Protects your CoCo III and Multi Pak Interface from destroying each other! Please specify MPI number 26-3024 or 26-3124 when ordering! Only \$9.95

In Quest of the Star Lord

(CoCo III only)

Enjoy the mixture of science and fantasy as you quest for the Phoenix Crossbow, the only thing that can save you in the post-holocaust world. 4 Disk sides of animated graphics adventure! Outstanding 320x200 graphics! Disk \$34.95 Hint Sheet \$3.95

Kung-Fu Dude

(CoCo I/II/III)

See Feb. '88 review. A Kung-Fu program for the CoCo. Destroy opponents and evade obstacles as you grow even closer to your ultimate objective! Disk \$24.95

PYRAMIX

(CoCo III only)

See Dec. '87 review. Brilliant colors, sharp graphics, and hot action! Disk .. \$19.95



AD&D Character's Companion

(CoCo I/II/III)

Great timesaving

utility helps create compatible AD&D characters. Includes dice rolling routine, pick ability, race & class. Buy from the Players Handbook, magic items & spell materials. Save, load, and print character info. 3 Disk sides \$24.95

White Fire of Eternity

(CoCo I/II/III)

See Dec. '86 review. A great graphics adventure! Disk.. \$19.95

CHAMPION

(CoCo I/II/III)

See May '87 review. Become a superhero in this action adventure! Disk.. \$19.95

Toll Free

1-800-441-GIME

Order Line

Technical assistance: 7pm to 9pm
Orders: 9am to 9pm Eastern time
On-line orders and up to date
information: Delphi's CoCo Sig

GIMMESOFT
P.O. Box 421
Perry Hall, MD 21128
301-256-7558 or 301-256-2953

Add \$3.00 for shipping and handling
Add \$2.50 for COD (USA only)
MD residents add 5% sales tax
VISA/MC/Check/Money Order/COD



FROM THE DESK OF...

Ted & Darlene Paul



First things first - a Happy Thanksgiving to all of our readers everywhere and a very Merry Christmas and Happy Chanuka as well.

Second, my apologies for being as late as we are with this edition. A number of advertisers, unable to get space in another CoCo magazine, asked us to make room for them. Knowing our readers are always looking for a good buy we said "sure." In addition we were also informed that "InterTan" (Tandy of Canada) will no longer be ordering CoCo 3's. We have received many letters from CoCo owners in Canada mentioning this to us. I spoke with Fran McGehee of Tandy and she said that this was solely an InterTan decision and that it in no way affected the CoCo in the U.S.A. She said that Tandy was fully behind the CoCo here in the States. Canadian CoCo owners who want to contact InterTan will find them in Barrie, Ont. The area code is 705. We urge you to write to let them know your feelings. Remember that you can catch more flies with honey than vinegar.

Other than that bit of bad news a LOT has been happening with *CoCo Clipboard Magazine*. We were the guests of the CoCo section of the Philadelphia Area Computers Society in September. Our thanks to Paul Sgrillo and his family for opening their home to us. Also our thanks to Bob Hengstebeck for making the arrangements and too the folks who attended the meeting at LaSalle College. Our thanks as well to Paul and Kathy Bornemann in Trenton, NJ for their hospitality on Saturday evening. Paul has written a couple of articles for us and is a very active business user of his CoCo's.



Dar draws a name for a free subscription.

October found us visiting many of you at the Hyatt Regency in Princeton, N.J. In particular our thanks to Tom and Jason of CoCo Tech for all of their help. Be sure to

check their ad in this issue. Also joining us was Jim DeStafeno and Ed Gresick of Delaware. Jim produces ClipDisk and Ed, who is using the CoCo to operate his Radio Shack franchise store, answered questions about programming and about the CoCo in general. Our thanks to all of you who stopped by and said hi.

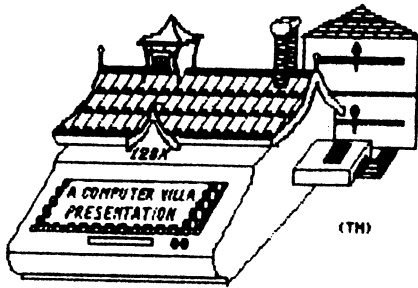
You may have noticed the number of our ads has increased. We can't thank our advertisers enough for their support. We can't urge you enough to support these hard working men and women as they support all of us with their hard work. And use those *Clipboard* coupons.

We've been asked several times if we would open a "Classified Ads" section in the magazine. We have resisted this for some time for a number of reasons. However more and more people have asked for this service and so we will open up a "Classified" section. We will be monitoring this experimental section very closely. The last thing we will tolerate is stolen merchandise and that includes pirated software. We reserve the right to refuse any ad and we will not take ads for x-rated, porno, or lottery/gambling software. Details are as follows:

- A) - All ads must be pre-paid in full. Check, money order or credit card. Credit card orders will be billed at 10% surcharge.
- B) - Ads are 50 cents a word with a 10 word minimum. Subscribers of record may take ads at 40 cents a word with a 10 word minimum.
- C) - Ads must be typewritten. They must be contain the name and full address of the advertiser and we must have your phone number for confirmation of your order.
- D) - *CoCo Clipboard Magazine* will not be liable in any way for items advertised in the Classified section. There are no refunds on these ads.
- E) - Classifieds are open only to private individuals or groups. Commercial ads will not be accepted.

This will be a limited 2 edition experiment. If we don't have any major problems we will continue the service.

Hope this finds you all well and having fun, producing and learning. Have a Happy New Year and we'll see you all in 1989.



"Software for your home since 1983"

1328 48th St.
Des Moines, IA 50311
515-279-2576

TERRY SIMONS' Computer Villa

P R E S E N T S

"Home-Pac 3.2"
F>A>S>T C>H>E>C>K/CARD & MAILER
+
DISK JACKET



128K, Disk - RGB or Composit Monitor Req.

Rush Caley of Emerson Computer Services says: "I have seen countless checkbook programs for the CoCo. Some were commercially available, others were written by users for their own use. In a word, I have NEVER seen a program for this purpose as good as Terry's Home-Pac."

1. Full 80 Column - large amount of data displayed
2. Credit-Card & Checks - Constant balance - two accounts
3. V3.2 has "prompt cards" making it easier then ever
4. Full sortability Plus - date parms - printer dump
5. Mailer Program & Disk Jacket Program - 70K+ CoCo Power
6. Make your CoCo a useful part of your house!
7. Home-Pac is "Satisfaction Guaranteed"!

See August Clipboard Review. **\$34.95**

Any order \$2.50 Shipping

*FREE Christmas Disk with any order!

By Bob Van der Poel All 32K CC2 or 3, disk Req.

1. Magazine Article Finder: Do you have over a year of CoCo Magazines? If you have one year or several years of CoCo magazines, "Indexer" will find article you want FAST! Telling you the name and month of the issue. Total of six years, including 1988. Covering over 5 CoCo Mags. including "Clipboard", and Rainbow of course. Over 200K of compressed data! (Two Diskfull). Plus it's a great a great magazine article indexer for; "any type file you'd like to create"! **\$21.95**

Three first class Programming Tools!

2. Disk Programming Pak: 4 routines a. ADDML appends ML routines to basis b. UNPACKER Undoes basic packed programs c. MLBASIC Converts a basic program to ML format to be LOADMed & EXECed. Encryps & BREAK protects, preventing others from viewing or altering your code d. JOIN Joins several ML routines, and/or adds auto-exec, if desired. **\$14.95**

3. PACKER: Removed REMed statements redesignating line No's. Removes spaces tokenizing basic to a more compact form. Joins lines up to 2K long! Five packing functions in all. +error checking **\$14.95**

4. MAPPER: Cross-references all Variables & Strings + GOTO's & GOSUB's, by line No. & occurrence times. Screen/Printer display. **\$14.95**

Van der Poel Utilities

5. Disk Management System: Auto-Program filing system also keeps copies of (and restores) disk directories. **\$14.95**

6. Disk Utility Pack II: Superfast - Format Backup 3 or less swaps per 64K system. Utilizes from 32K to your maximum 64/128/512K! Plus Find; Compare; Diskmatch; Occount; MENU 16 programs in all! **\$14.95**

7. BVDP Stock Manager \$19.95 8. BVDP "CoCo Three" Cribage \$13.95

Computer Villa, Christmas Graphic's & Carols
Graphics by Terry (g) Simons, are presently used by several
Top Software Co's. The Christmas disk Specify CoCo 2 or 3. \$4.95

The Machine Shoppe

Andrew Bartels

Have you tried to get started programming in Assembly, but not gotten very far? If you're used to BASIC programming, Assembly does have a complex look to it, and chances are there are MANY other CoCoists out there wondering about answers to the same Assembly questions you are! That's why CoCo Clipboard is printing my column, The Machine Shoppe - to answer your questions. So if you've got a problem dealing with Assembly, no matter how small or how big, drop me a line. I'll see if I can help you out in The Machine Shoppe. Mail your Assembly questions and troubles to:

Clipboard Machine Shoppe
c/o Andrew Bartels
1859 East 8th Street
Mesa, AZ 85203-6649
Phone: (602) 827-1934

If it's an emergency, and you just have to talk to me right away, I've included my "Assembly Questions Hotline" phone number here as well. Remember, no question is too big or too small. I look forward to hearing from some of you soon! Meanwhile, here is this two month's worth of questions and answers:

Q: What are the "Condition Codes"? I know they have something to do with the CC register, but since I'm just starting Assembly, I'm not sure what they do.

*Troy Gresick,
Springfield, Illinois*

A: The Condition Codes are a series of eight bits in the CC register. Those eight bits are called "Condition Codes" because they change often, and reflect the results of the last operation performed by the computer. A map of the bits in CC, along with what they do is in Table 1.

How do they work? Well, suppose you have written Assembly code to do a subtraction of two numbers. The result of that subtraction is recorded in CC. If the answer to the subtraction was zero, the "Z" bit of CC would be set (equal to one). The "N" bit of CC is set only if the result is negative. In the above example, "N" would be clear (equal to zero).

The "V" bit is set when an overflow occurs, such as when you add 70 and 200

Table 1: Bit Map of the Condition Code Register (CC)

Bit	Name	Description
0	C	Carry Code, for carry and borrow actions of adding and subtracting
1	V	Overflow Code, set when you overflow a register (like adding too much).
2	Z	Zero Code, set when two values are equal or a subtract equals zero.
3	N	Negative Code, set if one value is less than another, or if subtract results in negative number.
4	I	IQR Interrupt flag, clear when CoCo is on an IRQ interrupt call.
5	H	Half Carry, to carry out of bit 3.
6	F	FIRQ Interrupt, clear when CoCo is on a Fast Interrupt Request.
7	E	Entire State on Stack, to tell what a Return From Interrupt (RTI) does.

together, and try to keep the answer in only eight binary bits. The "C" bit is likewise set when there is a need to carry or borrow in addition or subtraction. Other bits in CC deal with coordinating the system interrupts so the machine does not lock up.

The relative branch instructions are prime examples of how you can make branches in your programs according to the conditions resulting from various tests. The BEQ instruction stands for "Branch if Equal". The branch actually occurs when the "Z" (zero) condition code is set. The computer performs a subtraction when it compares things, and a subtraction of two equal values will result in zero. So, you branch when two numbers are equal, the BEQ relies on the "Z" condition being set. The other conditional branches work in this manner, except with different CC register bits.

Take a look at chapter 6 of TRS-80 Color Computer Assembly Language Programming, by Bill Barden Jr., for more on how these branches work with the Condition Codes. You will find them to be a very important part of Assembly.

Q: I've been having trouble doing 8-bit signed multiplies. Fortunately, the 6809 instruction set has a MUL instruction that does 8-bit multiplies. But this is UNSIGNED. Is there a way to get signed values from it?

*Arlin Parsons,
Naples, Florida*

A: I'm sure you must then be familiar with the convention for signed numbers in binary, where the highest order bit determines the sign (one is negative, zero positive) instead of carrying a value. To get a signed multiply out of an unsigned one, just regard the high order bit as a sign, and not as part of the number.

First, the binary operation called Exclusive OR is what we can use to determine the resulting sign when the numbers are multiplied. Table 2 shows the truth table for Exclusive OR, in both binary, and in plus and minus signs (as it applies to mathematics). The listing of SMULT.ASM below shows how to do the signed multiply you asked about. Lines 110 to 140 get the resulting sign bit into bit 7 of a byte and push that byte on stack.

Table 2: Exclusive OR Truth Tables

Binary			Number Signs		
!	0	1	!	+	-
0	!	0	+	!	+
1	!	1	-	!	-

Lines 150 through 170 actually multiply the numbers to get the unsigned result. Now it's just a matter of clearing out any high order bit from the answer (taking it's absolute value) and then applying the sign bit we had saved earlier. Line 180 performs the absolute value, and line 190 logical OR's the sign byte on stack with the most significant byte of the answer. Since the sign in the sign byte is bit 7, and all other bits will be clear, we can easily put the correct sign on the answer with line 190.

Line 200 cleans up the stack, line 210 saves the answer to memory, and from there line 220 returns. It's that simple. You can always do this on signed numbers, regardless their size (assuming the size does not exceed the number of bits you are working with). This will even work on signed divides too. I hope this fixes the problem.

Q: I am a proud CoCo 2 owner, and up until now, I've had a cassette only system. I got into Assembly with the EDTASM+ program pak. I just got a new disk system, and was thinking it'd be interesting to see how to do disk I/O. Some Assembly applications just need that available.

*Dave Hillis,
Seattle, Washington*

A: You may have heard that disk I/O is much harder than cassette I/O. That's true, especially if your Assembly program communicates directly with the upper memory I/O bytes devoted to disk access. But that is

much too involved to discuss in a Q&A column such as this. Fortunately, TANDY provided us with a much easier way for this which is explainable quickly. It deals with a ROM routine called DSKCON. The address where DSKCON can be found is always, and will always be contained in ROM addresses \$C004-\$C005. The pointer to the parameter bytes for DSKCON operation is at \$C006-\$C007. All you do is LDX \$C006, and X will point to the correct place to store the necessary five parameters.

Assuming you've done a LDX \$C006, the following parameters are as shown in Table 3, relative to X. The first parameter is what operation you wish to perform. By putting a zero at the location referenced by ", X" you tell it to restore the drive read/write head to track zero. A code 1 there means no operation, 2 tells it to read a sector, and 3 instructs it to write a sector.

Now, the next step is to select the drive number. If you just bought a new disk system, you probably only have one drive to access, which would be drive number zero. But CoCo can support up to four drives (numbered 0-3). The drive number goes here at the address referenced by "1,X" (called DCDRV).

From here on, the parameters only need to be selected if you chose operation code 2 or 3 for DCOPC (read and write, respectively). This next value is referenced by "2,X", and is called DCTRK. It selects the disk track

== PRODEK SOFTWARE ==

c/o Mike Snyder, Rt. 2, Box 81, Allen OK, 74825

Adventures are supplied on Disk Only, but are transferable to tape. All are text only. CoCo 1, 2 and 3 compatible.

== Clipboard SPECIAL ==

With any order of 3 or more, you may choose a free program with the coupon on the coupon page!

Here's our line-up

- SHADOW WORLD Travel to the future!
- TREASURE SEARCH Find King Erik's Treasure!
- ASTEROID B7 Escape from an Asteroid!
- DESTINATION: UNKNOWN You are stranded!
- ROBOCROOK Catch the mechanical menace!
- COUNTRY CLUB Get There in Two Hours!
- SILENCE SYNDROME Find the Secrets of the Cloud!
- ANOTHER DIMENSION Escape to Another Time!

Plus more being created Every Day!

Please note: More than one game may be sent on the same disk on orders of two or more.

Just \$12.50 for each game ordered

STOCK MARKET PORTFOLIO FOR THE COLOR COMPUTER



If You Are Serious About Stocks...
This Program Is A Must!

Stock Market portfolio for the color computer will keep track of all your current stock listings and keep a listing of stocks you have sold by the year, they were sold with all totals, profit and loss, and percentages. More than one person can use this program as long as the first three letters on both first and last name are not the same. The program is menu driven and will give you the option for either screen print or information to be printed on printer.

Rush Check for \$22.00 plus \$3.00 shipping & handling to:
Papariz Enterprises
700 York St.
Williamsburg, VA 23185
Please allow 2-3 weeks for delivery
Sorry no C.O.D.S
VA residents add 4.5% sales tax.

Hard Disk Organizer (H.D.O.)

- <1> HDO can help you choose and execute your various application programs
- <2> You manke Menus and Scripts - HDO will do the work
- <3> If your system has the capacity, you can reach 2 million programs in just 8 keystrokes
- <4> Individuals with no knowledge of OS9 are able to use your programs
- <5> HDO has been successfully run on both the CoCo & 68K OS9 Systems
- <6> HDO supports multiple terminal configurations

Sample Menu:
Hard Disk Organizer System
1. Development of Files
2. Busi. & Word Processing
3. Communications Programs
4. Exit



The H.D.O. program is available from:
Bob Hengstebeck, 408 Grandview Ave.,
Feasterville, PA 19017, (215) 322-5455

H.D.O. is just \$24.95 + \$3.00 shipping and packaging.

number involved. The disk is divided into 35 tracks, so select anything from zero to 34 here.

Table 3: DSKCON Routine Parameters

Name	Relative to X, If You LDX \$C006	Description
DCOPC	,X	Operation Code: 0 = Restore to track zero. 1 = No operation 2 = Read 3 = Write
DCDRV	1,X	Drive number, from 0 to 3.
DCTRK	2,X	Track number, from 0 to 34
DCSEC	3,X	Sector Number, from 1 to 18
DCBPT	4,X	Address of 256 byte I/O buffer.

Next is DCSEC, referenced by "3,X", lets you select the sector number. Each disk sector contains 256 data bytes, and there are 18 of them on each track. So, choose a value from 1 to 18 to store here.

And lastly, we have DCBPT, referenced by "4,X". Here is how you tell Disk BASIC where your I/O buffer is located in memory. If you are writing, this is the address of the 256 data bytes that go out. If you are reading, this becomes the address at which Disk BASIC stores the 256 data bytes it reads from the disk.

Once you finally get done setting the parameters needed to specify what operation, what drive, where at on the disk, and where at in memory, you are ready to call DSKCON. To call it, just JSR [\$C004]. That jumps to the subroutine whose address is contained within \$C004-\$C005, where DSKCON's address is stored in ROM.

After you have called DSKCON, either what you told it to do is done, or things were aborted due to an error. So, you need to test out the error status report referenced by "6,X". It is called DCSTA. If that byte is zero, no errors have occurred. If some bit or bits in here are set, you've had some kind of error. Table 4 lists out the definitions of each bit of DCSTA. When an error happens, you'll have to provide the necessary routines to report the error.

Okay, now we know how to get one sector loaded in or written out. What about something more complex, like doing a DIRectory?

Well, we just use a series of DSKCON calls to read the directory, and from what we read in, we can then display a list of what's on the disk. The DDIR.ASM listing provides an example of how a double column directory can be printed to the screen, and how the repeated DSKCON calls are used to get the directory information.

This has been a fairly brief overview of disk I/O. Another version of explaining Assembly disk I/O is written in chapter 11 of your disk system owner's manual. That too will help you on your way to building the necessary I/O routines you'll need. The DDIR.ASM Assembler listing is a complete directory routine, and it will help you see how to use DSKCON to do the disk I/O while the rest of your program controls the process.

Table 4: DCSTA bit definitions

DCSTA is referenced at 6,X, where you've done a LDX \$C006. Bit error definitions are as follows:

Bit	Description
2	Lost data
3	CRC of data isn't right
4	Desired record not found
5	Write fault
6	Disk it write-protected
7	Drive not ready for use.

* The information for Tables 3 and 4 is taken from the disk system owners manual, Chapter 11: Technical Information.

Listing of SMULT.ASM:

```

00100          ORG      $E00
00110 SMULT    LDA      VALUE1  GET ONE VALUE
00120          EORA     VALUE2  FIGURE RESULTING SIGN IN BIT 7
00130          ANDA     #$80    KEEP ONLY BIT 7
00140          PSHS     A        SAVE SIGN BYTE FOR LATER!
00150          LDA      VALUE1  GET ONE VALUE
00160          LDB      VALUE2  GET OTHER VALUE
00170          MUL      DO UNSIGNED MULTIPLY
00180          ANDA     #$7F    GET ABSOLUTE VALUE (CHOP OFF HIGH BIT)
00190          ORA      ,S      PUT PRE-FIGURED SIGN ON IT
00200          LEAS     1,S     CORRECT STACK
00210          STD      RESULT  SAVE RESULT
00220          RTS      AND RETURN
00230 VALUE1   FCB      0       FIRST 8-BIT SIGNED NUMBER
00240 VALUE2   FCB      0       SECOND 8-BIT SIGNED NUMBER
00250 RESULT   FDB      0       SIGNED ANSWER (16-BIT)
00260          END      SMULT

```

Listing of DDIR.ASM:

```

00100 *DOUBLE COLUMN DIRECTORY
00110 *USING DSKCON
00120 *BY ANDREW BARTELS
00130 *FOR COCO CLIPBOARD MAGAZINE
00140 POLCAT EQU $A000
00150 CHROUT EQU $A002
00160 CLS EQU $A928
00170 ORG $E00
00180 PRINT PSHS A,X SAVE REGISTERS
00190 PRINT1 LDA ,X+ GET A CHARACTER
00200 [CHROUT] JSR [CHROUT]
00210 TSTA WAS IT A ZERO?
00220 BNE NO...THEN PRINT MORE
00230 PULS A,X,PC RETURN
00240 DDIR CLR SIDE,PCR SET SIDE TOGGLE TO LEFT
00250 LBSR COPY MOVE TEXT SCREEN TO BUFFER
00260 JSR CLS TO CLEAR SCREEN
00270 LDY $C006 POINT TO DSKCON PARAMETERS
00280 LDA #2 A=2
00290 STA ,Y SET TO READ
00300 LBSR GETGAT GO READ IN GAT FIRST OFF
00310 LDD #$1103 A=17,B=3
00320 STA 2,Y SET TRK = 17
00330 STB 3,Y SET SECT =3
00340 PSHS B SAVE SECTOR ON STACK
00350 LEAX BUFF,PCR POINT TO BUFFER
00360 STX 4,Y SET BUFFER TO DSKCON
00370 DDIR1 JSR [C004] CALL DSKCON
00380 TST 6,Y WAS THERE AN ERROR?
00390 BNE ERROR YES, THEN REPORT IT
00400 LDA ,X CHECK FIRST BYTE
00410 BEQ DDIR7 IF 0, FILE WAS KILLED...SKIP IT
00420 INCA IF IT WAS 255, IT IS NOW 0
00430 BEQ #8 IF IT WAS 255, THEN DIR IS DONE
00440 LDB #8 PRINT FILENAME
00450 LBSR NAM
00460 LDA #47 PRINT A "/"
00470 JSR [CHROUT]
00480 LDB #3 PRINT 3 CHARS
00490 LBSR NAM PRINT EXTENSION
00500 LDB 2,X GET FIRST GRAN #
00510 LBSR GRANUL GO PRINT # GRANULES
00520 LDA SIDE,PCR GET SIDE TOGGLE
00530 ANDA #1 IS IT ODD?
00540 BEQ DDIR3 NO, THEN PRINT SPACES
00550 LDA #13 YES, THEN PRINT <CR>
00560 JSR [CHROUT]
00570 BRA DDIR5 UNTIE PROGRAM FLOW
00580 LDA #32
00590 JSR [CHROUT] PRINT A SPACE
00600 INC SIDE,PCR NEXT SIDE
00610 LEAX 21,X POINT TO NEXT ENTRY
00620 JSR [POLCAT] WAS THERE A PAUSE KEY?
00630 BEQ DDIR6 NO...THEN KEEP ON
00640 LBSR PAUSE YES...THEN PAUSE FOR USER
00650 LEAU BUFF,PCR POINT TO BUFFER START
00660 LEAU 256,U POINT TO BUFFER END
00670 STU HOLD1,PCR SAVE
00680 LEAU HOLD1,PCR POINT TO POINTER
00690 CMPX ,U DONE WITH BUFFER?
00700 BNE DDIR2 NO...DO MORE IN THIS ONE
00710 LEAX BUFF,PCR YES...RESET BUFFER POINTER
00720 INC ,S NEXT SECT
00730 LDA ,S GET NEXT ONE
00740 STA 3,Y LET DSKCON KNOW

```

```

00750 DDIR7 BRA DDIR1 GO CALL DSKCON & DO IT OVER
00760 LEAX 32,X POINT TO NEXT ENTRY
00770 BRA DDIR6 CONTINUE
00780 ERROR LEAX MSG,PCR POINT TO MESSAGE
00790 LBSR PRINT PRINT IT ON SCREEN
00800 CLR $FF40 STOP DRIVE
00810 LEAX MSG1,PCR POINT TO MESSAGE#1
00820 LBSR PRINT GO PRINT IT
00830 BSR WAIT WAIT FOR A KEY PRESS
00840 BSR RESTOR RESTORE SCREEN THE WAY IT WAS
00850 PULS A,PC RETURN FOM DIR
00860 DONE $FF40 STOP DRIVES
00870 LDA SIDE,PCR WHAT SIDE IS IT ONE?
00880 ANDA #1 .S IT LEFT?
00890 BEQ DONE1 YES...SKIP PRINT
00900 JSR $B958 NO...FINISH LINE
00910 DONE1 LEAX MSG4,PCR POINT TO MESSAGE#4
00920 LBSR PRINT PRINT IT
00930 FREE GO PRINT FREE GRANS
00940 LEAX MSG1,PCR PRINT MESSAGE#1
00950 LBSR PRINT PRINT MESSAGE#1
00960 BSR WAIT WAIT FOR KEY PRESS
00970 BSR RESTOR SET SCREEN BACK TO NORMAL
00980 PULS A,PC RETURN
00990 NAM ,X+ GET A CHAR
01000 JSR [CHROUT] PRINT IT
01010 DECB DONE?
01020 BNE NO...PRINT MORE
01030 RTS RETURN
01040 LDX #1024 POINT TO TEXT SCREEN
01050 COPY LEAX BUFF2,PCR POINT TO BUFFER
01060 COPY1 ,X++ GET TWO BYTES
01070 STD ,Y++ SAVE IN BUFFER
01080 CMPX #1536 DONE WITH SCREEN?
01090 BNE COPY1 NO...COPY MORE
01100 LDX $88 GET CURSOR POSITION
01110 STX CADDR,PCR SAVE CURSOR ADDRESS
01120 RTS RETURN
01130 RESTOR LDX #1024 POINT TO SCREEN
01140 LEAX BUFF2,PCR POINT TO BUFFER
01150 REST1 ,Y++ GET TWO BYTES
01160 STD ,X++ RESTORE SCREEN
01170 CMPX #1536 DONE WITH SCREEN?
01180 BNE REST1 NO...KEEP ON
01190 LDD CADDR,PCR GETOLD CURSOR ADDRESS
01200 STD $88 RESTORE CURSOR
01210 RTS RETURN
01220 WAIT [POLCAT] GET A KEY
01230 BEQ WAIT NONE...KEEP WAITING
01240 RTS RETURN
01250 PAUSE A,X
01260 LEAX MSG2,PCR POINT TO MESSAGE#2
01270 LBSR PRINT PRINT IT
01280 PAUSE1 [POLCAT] GET A KEY
01290 BEQ PAUSE1 IF NONE...THEN WAIT MORE
01300 LEAX MSG3,PCR POINT TO MESSAGE#3
01310 LBSR PRINT PRINT IT
01320 PULS A,X,PC RETURN
01330 MSG $D0D 2 CHR$(13)'S
01340 FCC *INPUT/OUTPUT ERROR!*
01350 FDB $D00
01360 MSG1 $D
01370 FCC /PRESS ANY KEY.../
01380 FCC /<PAUSED>/
01390 MSG2 0
01400 FDB $808
01410 MSG3 0
01420 FDB $808

```

```

01430 FDB $808
01440 FDB $808
01450 FCC 0
01460 MSG4 /FREE GRANULES=> /
01470 FCC 0
01480 CADDR 0
01490 CURS 143
01500 COUNT 128
01510 SIDE 0
01520 HOLD1 0
01530 GRANUL PSHS
01540 LDA D,CC,X SAVE REGISTERS
01550 STA NUMGRN,PCR RESET GRAN COUNTER
01560 LEAX GATBUF,PCR POINT TO BUFFER
01570 ABX GET ADDR OF GRAN INFO
01580 LDA ,X GET INFO
01590 STA GRNINF,PCR SAVE IT FOR LATER
01600 ANDA #80 IS THIS THE LAST ONE?
01610 BNE GRAN2 YES...THEN PRINT & STOP
01620 LDB GRNINF,PCR NO...GET INFO
01630 INC NUMGRN,PCR COUNT ANOTHER GRAN
01640 BRA GRAN1 GO DO IT AGAIN
01650 GRAN2 #32 FORMAT
01660 LDA [CHROUT]
01670 JSR
01680 CLRA
01690 LDB NUMGRN,PCR GET # GRANS COUNTED
01700 PSHS D
01710 JSR $BDCC PRINT D
01720 PULS D
01730 CMPB #10 DO WE NEED TO FORMAT MORE?
01740 BGE GRAN3 NO...SKIP
01750 LDA #32 YES...FORMAT WITH SPACE
01760 JSR [CHROUT]
01770 PULS D,CC,X,PC RETURN
01780 LDD #$1102 TRK=17, SECT=2
01790 STD 2,Y SAVE FOR DSKCON
01800 LEAX GATBUF,PCR POINT TO GAT BUFFER
01810 STX 4,Y
01820 JSR [C004] GET GAT FROM DISK TO BUFFER
01830 TST 6,Y WAS THERE AN ERROR?
01840 BEQ GETGT1 NO...THE RETURN
01850 LEAS 1,S HALF-WAY FIX STACK
01860 LBRA ERROR GO REPORT ERROR & STOP
01870 RTS RETURN TO DDIR
01880 NUMGRN 0
01890 FCB GATBUF,PCR POINT TO GAT TABLE BUF
01900 LEAX #68 GET # GRANS TO CHECK
01910 LDB B SAVE ON STACK
01920 PSHS ZERO FREE GRAN COUNTER
01930 CLR B GET A BYTE
01940 LDA ,X+ IF IT WAS 255, IT IS NOW 0
01950 INCA FREE2 IF NOT ZERO NOW, SKIP
01960 BNE IF IT IS, ANOTHER GRAN FREE
01970 DEC ,S WAS IT THE LAST ONE?
01980 FREE1 NO...COUNT MORE
01990 BNE YES...PRINT FREE GRANS
02000 CLRA LIFT STACK
02010 LEAS 1,S
02020 JSR $BDCC
02030 RTS $B958
02040 RMB 256
02050 BUFF RMB 512
02060 BUFF2 RMB 256
02070 GATBUF RMB 256
02070 END DDIR

```

Published Bi-Monthly

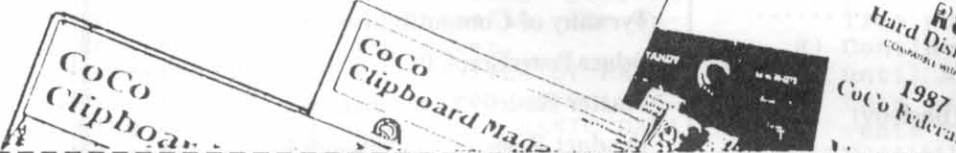
CoCo Clipboard Magazine

THE NEWEST, MOST INDEPTH MAGAZINE FOR TANDY'S COLOR COMPUTER 2 & 3

If you've been looking for a CoCo magazine that takes a No Nonsense approach to the Color Computer, then CoCo Clipboard Magazine is for you!

If you're getting tired of lo-res block graphics applications, reviews that still leave you guessing, and the same old boring approach to serious computing, then CoCo Clipboard Magazine is for you!

Clipboard brings you solid articles, programs and tutorials on BASIC, OS9, BASICØ9, "C", MultiView, Machine language, databases, packet radio - the list goes on and on. Plus we're up to date on what's happening with the CoCo. We were the first to bring you IN DEPTH reviews on VIP Writer III, RGB Hard Drives and Tandy Home Publisher and every month we sponsor an on-line conference on CompuServe.



6 - 2 - 1
\$83.25
SAVE 15% FROM LIST PRICE

Just in time for holiday giving it's 6-2-1! "6-2-1" is our new program to save your money! Six back issues of *Clipboard*, a two-year subscription to *Clipboard* and one year of *ClipDisk*. Normally \$97.97, now you can save 15%, only \$83.25! Just write in "6-2-1" on the order form on page 18.

MC - Visa - Discover accepted.

I am: _____ Subscribing
_____ Renewing
_____ Gift Subscription

Name: _____

Address: _____

City: _____

State/Prov: _____

Zip/Postal Code _____

- ★★ _____ 1 year of Clipboard \$15.00 U.S. Funds _____
 - _____ 2 years of Clipboard \$28.00 U.S. Funds _____
 - _____ 1 year Canadian \$18.00 U.S. Funds _____
 - _____ 1 year Overseas \$27.00 U.S. Funds _____
 - _____ Single Issue ClipDisk \$ 9.95 U.S. Funds _____
 - _____ Specify Month _____
 - _____ 1 year of ClipDisk \$49.95 U.S. Funds _____
 - ★★ _____ Back Issues 2 thru 7 \$20.00 U.S. Funds _____
 - _____ 6-2-1 Special \$83.25 U.S. Funds _____
- TOTAL** _____

Please charge my purchase to my (circle) VISA / MasterCard / Discover Card.

MasterCard Interbank Number

Card # _____

Exp. Date _____

Signature _____



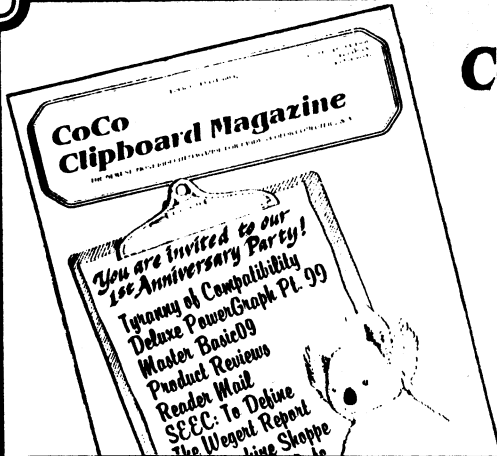
PLEASE NOTE: All charge card orders are handled courtesy of On-Line Computer Services. Your charge card billing will show On-Line Computers and NOT CoCo Clipboard Magazine.

CoCo Clipboard Magazine
3742 U.S. 20, Box 3, Fredonia, NY 14063

★★ **ONLY \$35.00** ★★

To SUBSCRIBE and
receive **6 Recent**
BACK ISSUES

Clipboard brings you solid articles, programs and tutorials on BASIC, OS9, BASIC09, "C," MultiView, Machine language, databases, packet radio - the list goes on and on. Plus we're up-to-date on what's happening with the CoCo. We were the first to bring you IN-DEPTH reviews on VIP Writer III, RGB Hard Drives and Tandy, Home Publisher and every month we sponsor an on-line conference on CompuServe.



CoCo Clipboard Magazine BACK ISSUES

Our 6 Most Recent Issues for only \$20.00

Vol. 1 Issue 5 May/June 1988

- DataBase Tutorial Pt. III**
by Rush Caley
- The Basic Idea**
by Randy Krippner
- Master Basic09**
by Bill Brady
- Basic Help**
by Bill Bernico
- CoCo 'N Amateur Radio**
by Mike Dooley KE4PC
- Quick Forecasting**
by Jim DeStafeno
- Line Input with Editing (Basic09)**
by Bob van der Poel
- Product Reviews**
by Ralph Dahlgren and Mike Dooley
- The "Wegert Report"**
by Steve Wegert

Vol. 1 Issue 6 July/August 1988

- The Wegert Report** by Steve Wegert
- CoCo and Small Business** by Terry Simons .
- CoCo 'N Amateur Radio** by Mike Dooley KE4PC .
- Master Basic09** by Bill Brady
- New Boots** by Bob van der Poel
- The Assembly Line** by Kraig Brockschmidt .
- Programming Challenge** by Ted Paul
- The Machine Shoppe** by Andrew Bartels ...
- Product Reviews** by Clipboards Review Crew .
- "SEEC" for "C"** by Nancy Ewart
- Deluxe Power Graph** by Randy Krippner ...

Back issues of CoCo Clipboard are always available. The price is \$2.75 per issue (U.S. funds please). There is a \$1.00 p/h charge for the first issue ordered and \$.50 p/h for each additional issue purchased.

And now you can phone in your order! We accept VISA, MasterCard and Discover - (716) 679-0126 - 9 a.m. to 9 p.m. Monday thru Saturday!

Vol. 2 Issue 1 Sept./Oct. 1988

- CoCo 'N Amateur Radio** by Mike Dooley KE4PC .
- Tyranny of Compatibility** by Rush Caley ..
- Deluxe PowerGraph Pt. II** by Randy Krippner .
- Master Basic09** by Bill Brady
- Product Reviews** by Clipboards Review Crew
- SEEC: To Define** by Nancy Ewart
- The Machine Shoppe** by Andrew Bartels ...
- Modular Programming** by Boisy Petre
- The Wegert Report** by Steve Wegert
- What's In A Date** by Bob van der Poel
- Deluxe Power Graph** by Randy Krippner ...

Also Available

- Vol. 1 Issue 1 Sept./Oct. 1987** Photocopy
- Vol. 1 Issue 2 Nov./Dec. 1987** Photocopy
- Vol. 1 Issue 3 Jan./Feb. 1988** Photocopy
- Vol. 1 Issue 4 March/April 1988**
- Vol. 1 Issue 5 May/June 1988** Photocopy
- Vol. 1 Issue 6 July/August 1988** Photocopy

The Assembly Line

Kraig Brockschmidt

Welcome once again to the Assembly Line. As of October, 1988, feel free to address questions to me on CompuServe through my ID, 76701,76.

ML-DATA Turbo Transform:

At one time I needed to put 2028 bytes of machine language into Basic DATA statements so the ML could be poked in memory by a Basic program. Well I wasn't very ambitious to type 2028 numbers so I made a 111 byte ML program that did it for me. I call it ML-DATA Turbo Transform.

The binary program for ML-DATA Turbo Transform can be stored in memory using the Basic program in Listing 1 (ML-DATAB.BAS) or by assembling the source code in Listing 1A. Once the program is in memory, (C)SAVEM "ML-DATA",&H7F91,&H7FFF,&H7F91. Listing 2 (ML-DATAO.BAS) is an operating program for ML-DATA which facilitates use of this utility. Cassette users--change the LOADM in line 1 to CLOADM "and CSAVE the operating program before the ML program on a cassette. The operating program is described later.

As in previous articles, a line of BASIC code stored in RAM has a rigid structure. Transferring machine language to BASIC DATA statements requires us to build this structure. For a disk system, BASIC programs start at address \$2600 if 4 graphics pages are reserved (PCLEAR 4). A BASIC program in memory appears as follows:

```
$2600 :Always 0
$2601-02 :A 2 byte address pointing to
         the data for the next BASIC
         line.
$2603-04 :A 2 byte value for the line
         number.
$2605-   :Data for the BASIC line with
         all commands tokenized
         (see Vol 1 #2 of CoCo Clipboard).
```

At some point there is a 0 indicating the end of the BASIC line. The next byte in memory is the start of the next line. The address of this next byte is the address stored at \$2601-02. From this point on, the structure repeats for each line of BASIC code, until three zeros occur, signaling the end of the program.

Through the following steps, ML-DATA builds BASIC lines. The numbers in par-

enthsis indicate the lines of the assembly code where the step is done:

- 1) Determine where to begin forming Basic lines. (110-130)
- 2) Get the starting line number and the start address for the ML to be transferred. (140-160)
- 3) Write the line number in the current Basic line. (170)
- 4) Write a "DATA" and a space the current line. (180-190)
- 5) Check for the end of the ML being transferred" and exit a loop if it has been reached. (210-220)
- 6) Otherwise get a byte of ML, convert it a two-byte ASCII string (with hex. digits), store it in the DATA statement, and add a comma. (200-480)
- 7) If the DATA statement is longer than desired, start a new line. To start a new line, clear the last byte, store next address at the beginning of the previous line, and increment the line number. (410-520).
- 8) Continue loop by returning to step 3 until step 5 exits loop. (530)
- 9) Upon exit, a termination routine prevents the end of the last line from having just a comma or just a line number with "DATA".

It also stores the new end address of the Basic program at \$1B and returns to Basic. (540-660)

When the operating program is used, the machine language program automatically loaded. You are asked to enter the start and end addresses of the ML to be transferred and the line number you wish the DATA statements to begin at. These values are stored in \$14C, \$14E, and \$150 respectively. Note that the start address should always be less than the end address and the starting line number greater than 6. If the start > end you will get no DATA statements. If your line number is > 65535 you'll get an ?FC ERROR IN 5. Also, if you specify a large beginning line number, the line numbers may wrap-around to low numbers at the end of the BASIC program. Continuity problems can be fixed with a RENUM.

As always, there are several points of interest in the ML program. The main routine (200-480) loads a byte of ML data to transfer into the A register and copies it to B. If

you are familiar with hexadecimal digits stored in a byte, you will know that the upper 4 bits (the MSN, Most Significant Nibble) contain the first digit, and that the lower 4 bits (LSN, Least Sig. Nibble) contain the second digit. We must isolate and individually convert these digits to ASCII.

Lines 240-290 isolate the first digit in the A register by performing a logical AND with a bit pattern of 11110000. The last 4 zeros mask out the second digit. Similarly the second digit is isolated in the B register through a logical AND with 00001111. The bits of the A register are then shifted four bits to the right, so it contains the actual value of the digit. For example, the conversion of the value \$C8 into two digits proceeds as follows:

Line	A register	B register
230 TFR A,B	11011000=\$C8	11011000=\$C8
240 ANDA #\$F0	11010000=\$C0	11011000=\$C8
250 ANDB #\$0F	11010000=\$C0	00001000=\$08
260 LSRA	01101000=\$68	-
270 LSRA	00110100=\$34	-
280 LSRA	00011010=\$1A	-
290 LSRA	00001101=\$0C	00001000=\$08

The A register now contains the value of the first digit, \$C, and the B register contains the second, \$8. Lines 300-390 then converts both registers to ASCII. In line 300, A is compared to 9. If it is greater than 9, then it must be a hexadecimal letter (A-F) so we branch to LTR. If not, we add 48 decimal to the value giving it a possible range of 48-57 decimal. These are the corresponding ASCII codes for "0"-"9". If we branch to LTR we add 55 decimal instead. Since it was originally 10-15, the possible values for A after the addition are now 65-70, which translates to "A"-"F" in ASCII.

This process is performed on both A and B registers (lines 300-390) and then the two-byte ASCII value in the D register (A and B together) is written to the current BASIC line under construction (400). Lines 410-450 check the length of the line under construction, we first save the value of X in D and then subtract 77 from X. This number (77) corresponds to the length of a BASIC DATA statement with 25 elements. We then compare the subtracted value to the beginning of the line stored at S. If X-77 is less than the value at S, a comma is stored after the data element and we continue building. If not, there are 25 pieces of data in the current line so we branch to EOL, starting another line. Note that the TFR D,X instruction in line 440 does not effect the CC register.

Lines 490-520 finish up the current line and starts the next. It clears the last byte of the line, stores the next location at the beginning, of the line (see structure), stores this value at S, and adds 5 to the line number counter (in U). Line 530 jumps back to line 200 where we continue building

Basic lines. When we reach the end of the ML to be transferred the program exits the loop to FINUP (540-660). This routine simply fixes the end of the Basic line so there are no commas or line numbers ending the line. It also stores the end address of the new Basic program at \$1B and returns to Basic.

By the way, if you're wondering why this utility is dubbed 'Turbo Transform', try this: enter START=&H8000 and END=&H9FFF in the operating program. LISTing the result will take a while, but prepare to wait a whole 0.93 seconds for ML-DATA Turbo Transform to construct it!!

```

10 '*****
11 '* ML TO BASIC UTILITY *
12 '* COPYRIGHT (C)1988 *
13 '* BY KRAIG BROCKSCHMIDT *
14 '* 14506 NE 37th Pl. F-8 *
15 '*Bellevue, WA 98007-3486*
16 '*****

20 CLS: CLEAR200, &H7F91: FORT=1TO1
11: READA$: X=X+VAL("&H"+A$): NEXT:

```

R.A.D. PRODUCTS

presents

Textform



TEXTFORM is compatible with all Color Computers with a minimum of 64K, disk drive, and printer. This machine language program will format ASCII text files into two column pages quickly and easily. This is not another word processor. TEXTFORM is a user defineable two column text formatter program which will allow your Color Computer to create very professional documents without hours of tedious work. TEXTFORM is a versatile enhancement to any word processing system whether you are a casual or professional user.

Ideal For:

- Bulletins
- School newspapers
- Advertisements
- Club newsletters
- Program Listings
- Business reports
- And much more..

System Requirements:

CoCo (64k minimum), Disk drive Printer

Holiday Special ---> \$29.95!

R.A.D. Products
194 Hotchkiss St., Jamestown, NY 14701
(716) 665-2124

Terms: Check, C.O.D., Money Order. New York residents add 7% sales tax. C.O.D. orders add \$3.00. All orders add \$3.00 for shipping.

see our coupon on the coupon page

Using OS9 Files

Bob van der Poel

As I've been telling you all for some time now, OS-9 is a super operating system for the Color Computer. But something we all tend to forget from time to time is that OS-9 is a DISK operating system. Not only does this mean that OS-9 can load and save programs or data to/from disks, it also means that the system itself thrives on all kinds of disk operations. But before we get to this month's article which shows one way we can make BASIC09 do some mundane work for us, let's review OS-9's file structure.

OS-9, unlike some other operating systems you may be aware of, has only one type of file. There is no inherent difference between the structure of a file containing a list of names and addresses or the file containing the program which reads it. OS-9 cannot tell whether a file contains data or executable code -- that is job of file attributes. Note that attributes are not a part of the file, they are part of the file descriptor.

The file descriptor is a one sector block of data which contains -- among other things -- the attributes of the file, the date the file was created and last modified, and information on where the file is located on the disk. Whenever OS-9 accesses a file this information is read. Depending on who you are and what you are trying to do OS-9 may or may not process your request. On a multi-user system you may not be the owner of a file, and unless the "public read" attribute is set OS-9 will not let you read the file. Similarly, if the "write" attributes are not set no one can write or delete the file. If a file is an executable program then it will have the "execute" attributes set. Again, there are separate "execute" flags for the owner of a file and other users on the system (public). Most of the file attributes can be changed with the OS-9 ATTR utility, however you cannot change the attributes of a file you do not own.

File attributes cause lots of problems for beginners -- especially if they have downloaded a program, attempted to create a new Shell module by merging several programs together, or extracted a program with an archiving utility (like AR). OS-9 will balk at any attempt to load a program which does not have its "execute" attribute(s) set. It's simple to do -- just use ATTR to set the "E" and "PE" attributes. Does this mean that if we change the attributes of a data file to "E" and "PE" it will be loaded into memory?

No. OS-9 does other checks to make sure the "program" is correct...but that's grist for another article.

Most of the time when we think about files we focus on two kinds: Data files and executable programs. But let's not forget that files can be used for more than just data storage. Files can also contain instructions for programs or OS-9. Procedure files are an example which comes to mind right away.

A "procedure file" is a file which contains commands which you would normally type in from the keyboard...but because we're lazy (or just against drudgery) we can put these instructions in a file. The file "startup" on your boot disk is a good example of a procedure file. Startup contains instructions which OS-9 fulfills each time it is booted. The only unique thing about "startup" is its name -- OS-9 always checks the root directory of the disk it is booting from for a file called "startup." If it finds one, the instructions in the file are carried out. If "startup" does not exist, no damage is done. One caution about invoking programs from within procedure files: When interpreting a procedure file OS-9 expects input to come from the file, not the keyboard. If a program needs input from you (the keyboard) then you must redirect input when the program is invoked.

Here's a quick example. You wish to automate the process of creating a new system disk, so you write the following procedure file:

```
echo Creating new boot disk, ready /d1
format /d1
cobbler /d1
mkdir /d1/CMDS
chd /d1/cmds
copy /d0/cmds/grfdrv grfdrv
copy /d0/cmds/shell shell
```

The above procedure will not work. Format will print an "are you sure" message and wait for you to answer y(es) or n(o). Since input is coming from the file the first reply will be "c" (the first letter in "cobbler"). Format will re-prompt and get "o" as a response. Eventually it will get to the the "y" in "copy" and the disk will be formatted. However OS-9 will not like the name "/d0/cmds/grfdrv," nor will it be able to execute the command "grfdrv."

The solution? Well actually there are two. We could change the format command so that no keyboard input is required:

```
format /dl r "MyDisk"
```

will work fine. However, if we wish to have a prompt -- nice, just in case the disk in /dl contains valuable data -- then we could use:

```
format /dl </1
```

Note that instead of "</1" we could have used "</term," but that would only work if we were using "term" as our active window. If would create all kinds of problems if we were using "/w2."

If you create a special "startup" file to automatically invoke a favorite program, then redirection is also needed. I have a special disk which brings up my mailing list program. The final line in the startup file reads:

```
dml9 <>>>/1
```

Here not only input, but also output and error output has been redirected. The last two should not be necessary, but leaving them out sometimes causes problems. Play it safe and redirect all three paths.

There is one more use of files: communicating with other programs. Probably the best example of this is the C-Compiler executive. Creating a C program is a rather complex process of creating/editing a text (source) file, having various programs convert the source file to an assembly language source file (compiling), calling RMA to assemble the complied code, and then calling the OS-9 linker to create the final program. Rather than forcing you to remember all this the C-compiler comes with a program which creates a procedure file with all the necessary instructions to convert your source file to a loadable program.

I write a fair number of letters. I use an editor to create the text of the letter, save the text to a disk file, then I use one of the many text formatters available for OS-9 to format the text and print it out. Simple, but a bit tedious. First off, there's that matter of my "letterhead" information. I keep a file called "header" which is included in each letter. It contains my name, address, etc. and some formatting information. However, it must still be loaded (or included) for each letter I write. But a greater irritant is the fact the the formatter will only print one copy of the letter. If I want two copies I have to call the formatter twice. And if I want the second copy to be in correspondence quality rather than draft then I have to remember to set the printer. Too many things for me to remember. But aren't computers supposed to be good at doing tedious things?

The program listing "lp" takes the tedium out of printing letters. It's a simple pro-

gram which I hope you will change and expand for your own needs.

lp creates a file which contains the following:

1. Some initial format information for my text formatter,
2. The letterhead information with my name, etc.,
3. A line to print out today's date,
4. An "include" instruction to print the actual letter,
5. A closing to the letter,
6. A "start new page" command,
7. A format code to turn on correspondence mode,
8. A duplication of items 1 to 6.

After the file is created the program calls the formatter and two copies of my letter are printed. Finally, the temporary file is deleted -- all this simply by typing the command:

```
lp ("jim")
```

Note the parentheses and quotes around the argument. These are needed for a packed BASIC 09 program, unless you are using an enhanced shell program such as Shell+.

There is nothing very complicated about lp, however I do wish to point out a few high-



NEW

Broke 'r Broker

by Roy C. Pierce (c) 1986

Buy & Sell your way to Power in this Exciting Stock Market Simulation.
2 - 6 Players Ages 10 & Up.
"Not since Stock Ticker® have I had so much Fun!" F.G. Dawson
"Easy to Learn, Fun to Play!" A.R. Fazzackerley

128 K CoCo Required, Disk or Cassette. **\$23.95**

GOOD GAMES TRIO

Challenging Two Player Games



by Roy C. Pierce (c) 1988



ADI



OTHELLO

CONNECT 5

FAST AND FUN FOR ALL AGES
EASY TO RUN
ALL BASIC COMPLETELY LISTABLE **\$19.95**



HELLO/BAS

(See September & October Rainbow)

Disk Directory Utility

"...One Professional Looking, Well Behaved...and User Friendly Utility for BASIC Programs!"
"Do your Fingers a Favor and Introduce the CoCo in your Life to HELLO/BAS." October Rainbow Review

\$19.95

Prices Shown are in U.S. Funds.
Include Cheque or Money Order when Ordering.
On Orders of 2 or more Programs Deduct 10% from Total Price.
We Pay the Freight
In Canada Please Phone for Prices.
ALL Foreign Orders Add \$5.00
Sorry No C.O.D.

RCP

PIERCE

SOFTWARE

P.O. BOX 1787,
Main Post Office,
Edmonton, AB, Canada
T5J-2P2
PH: (403) 474-8435

lights. The routine which builds the string "today" assumes that the system date has been properly set. The month name is determined by reading through the data statements for the value of the month number from DATE\$. We extract the month number with the MID\$ function. The IF/ELSE is needed to strip off a leading "0" from dates like "January 03."

Before attempting to create the temporary file, we first attempt to delete that same file. In most cases this will not be necessary, however if we attempt to create a file which already exists we'd have to delete it anyway. Note that error trapping is only active for the DELETE. Any other errors will simply end lp.

The subroutine at line 100 reads the data statements at 1000 and sends them to the temporary file. Have a good look at the data statements -- not only do they contain string constants, they also contain variables and expressions. BASIC09 is the only language I know of which permits expressions in data statements -- a very useful feature.

As it stands, lp is a very specific program -- something I don't like to publish. However, it should be easy enough to modify to include your letterhead and codes for your text formatter. Feel free to play with the code -- that's how you learn.

Finally, this months challenge: Can you add or devise a "better" method for creating the

temporary file name? Can you find another use for a program like lp? Let me know, maybe you'll even get your name mentioned here! You can write to me at:

17435-57 Avenue
Edmonton, Alberta
Canada T6M 1E1

*Join Ted Paul & Dan Robins
for our monthly Clipboard conferences on CompuServe! Please note: we've moved the Conferences to FRIDAY NIGHTS! The time is the same 9pm Eastern.*

The November guest will be Tony DeStafeno from Disto. Our December guest will be Bill Vergona from Cer-Comp and January will tentativley feature Dr. Lester Hands - author of Lyra.

PROCEDURE lp

```

0000
0001 (* lp--Letter Printer
0016 (* An xprint interface for correspondence.
0017
0041 (* This program will do the following:
0042 (* 1. create a file called lp.temp,
0068 (* 2. initialize the printer,
0088 (* 3. call xp to print file.
00A8
00C4 (* lp.temp will contain the following:
00C5 (* 1. my letterhead,
00EB (* 2. today's date
00FF (* 3. xp include for letter to be printed
0111 (* 4. xp inst. for empasized print
013A (* 4. items 1...3
015C
016C (* Copyright 1988, Bob van der Poel
016D
0190
0191
019D PARAM FileToPrint:STRING[50]
019E
019E DIM A$,Today:STRING[100]
01AE DIM TempName:STRING[40]
01BA DIM Path:BYTE
01C1 DIM T:INTEGER
01C8
01C9 (* create today's date
01CA
01E0
01E1 FOR T=1 TO VAL(MID$(DATE$,4,2))
01F7 READ Today
01FC NEXT T
0207 Today=Today+" "
0213 IF MID$(DATE$,7,1)="0" THEN
0223 Today=Today+MID$(DATE$,8,1)
0232 ELSE
0236 Today=Today+MID$(DATE$,7,2)
0245 ENDIF
0247
0248 Today=Today+" ",19"+LEFT$(DATE$,2)
025C
025D DATA "January","February","March","April","May","June","July"
029A DATA "August","September","October","November","December"
02D3
02D4
02ED GOSUB 50 \>(* initialize printer
02EE
02EE IF FileToPrint="" THEN
02FA PRINT
02FC PRINT "lp--Letter printer interface for xprint"
02F7 PRINT "Usage: lp filename"
033D PRINT
033F END
0341 ENDIF
0343
0344 (* create a temporary file for xp.
0366 TempName="lp.temp."+FileToPrint
0367
037A ON ERROR GOTO 10
037B DELETE TempName
0386 10
038A ON ERROR
038D CREATE #Path,TempName:WRITE
0399
039A (* stuff for 1st copy
03AF
03B0 GOSUB 100

```


Clip-Ettes

Jim DeStafeno

- LINE LINEUP:

This one is from Rush Caley as a result of my complaint. He laughed when he found I didn't know. Was I the only one in the dark? As anyone knows that uses a CoCo word processor, the printer code uses a character space on the screen, but since the printer doesn't print them, the printed line is one character shorter than the screen line.

A printed line with 4 or more printer codes looks bad enough; pretty short. However, if you make a form with vertical lines, bold print, underlining, etc., lining up columns can become a nightmare.

Try this, it works perfectly. The base of the idea is founded on the fact that most every printer code is, or can be, preceded or followed by a space. So, just set a space in the printer code itself and don't use one in the text. Then when the printer encounters a printer code, it will not print the code, and thereby lose a character space in the printed line, but it will print a space. The result is a net gain/loss of zero line character spaces between the displayed line and the printed line, and therefore everything lines up on the screen and on the printed page.

The drill goes this way. When you set up the printer code, put in a 32 either in front of or behind the code. (That is the ASCII value of a space.) Example, 32 nn nn nn or nn nn nn 32. (The ns represent the printer code.) When you type in a line, think of the printer code as a space; do NOT type in a space.

- FUNCTION, CTRL and ALT KEYS SENSING FROM BASIC:

The program below demonstrates how to sense the function keys from BASIC. It's not a bad input engine for fields or a word processor. (Note, if you do use it, be sure to filter out the Left Arrow and Clear keys; similar to what is done in line 20 for ENTER. You'll find the program to be fast enough, but the Hi-Speed Poke wouldn't hurt. However, a fist full of IF statements will slow it down. It will work fine if a second INSTR statement is used to do the filtering.)

Anyway, check out how this thing works. There is a keyboard table in memory locations 338 through 345. If the memory location has a number in it other than 255, a key is being pressed. A given number in a given location tells the computer which key is being pressed. The location for <F1> is 343; <F2>

is 344, CTRL is 342 and ALT is 341. The For/Next lets us Peek into these locations and check out the values.

```
5 CLS:C$=CHR$(255):C=191:D$=CHR$(103)+CHR$(52)+CHR$(4)+CHR$(214)+CHR$(189)+CHR$(1)+CHR$(64):PRINT@32,C$;
10 EXEC44539:I$=INKEY$:FOR X=341 TO 344:
IF PEEK(X)=C THEN ON INSTR(D$,I$) GOTO31
,32,33,34,35,36,37 ELSE NEXT
20 IF I$=CHR$(13) THEN A$="":CLS:GOTO5 ELSE A$=A$+I$:PRINT@32,A$C$:GOTO10
31 PRINT@452,"<F1> KEY ACTION":GOTO10
32 PRINT@452,"<SHIFT F1> KEY ACTION":GOTO10
33 PRINT@452,"<F2> KEY ACTION":GOTO10
34 PRINT@452,"<SHIFT F2> KEY ACTION":GOTO10
35 PRINT@452,"<CTRL> KEY ACTION":GOTO10
36 PRINT@452,"<SHIFT CTRL> KEY ACTION":GOTO10
37 PRINT@452,"<ALT> KEY ACTION":GOTO10
```

It is handy each of key's "is being pressed" code is 191. So if we Peek the four locations and find a 191, we know which key is being pressed. However, the shift key has no effect on the numbers in keyboard table. (Expect there is another memory location to sense that, but I don't know where it... Help?)

So once the program has determined if one of the four keys has been pressed, INSTR is used to determine which one. Notice line 5 sets D\$ to the ASCII code, CHR\$(n), of seven characters, our four keys; shifted and unshifted. (The shifted ALT doesn't affect the key table.)

One question might come to mind, "Why not just use the ASCII code to sense which key is being pressed without using the For/Next?" Well, F1 has the same ASCII code as the overcase "g", one of the others is the same as "4" and CTRL-ALT is the same as "@"; but one of them have the same "key table code" as their counterpart.

Of course if you want to get fancy, there are lots of ways to modify the program to change the quantity and/or order of the sensors to get different results.

YOU and CLIPETTES:

If you have some favorite programming ricks, send them in to the magazine or myself; RD1 Box 275, Wyoming, DE 19934.

GOT THOSE

"Compatibility Blues?"

You're not alone. Many Color Computer users have experienced the frustration of buying a cheap imitation of our Hard Drive Basic and then discovering that their favorite programs *no longer work!*

Now, you can **FIX** those compatibility problems, **ELIMINATE** the bugs, **SPEED** up your system and **PREPARE** for tomorrow's disk drives . . . buy your DOS from the company that **invented** the original Hard Drive Basic years ago!

Supercharge your Color Computer with RGB-DOS!

Why RGB-DOS? Because RGB-DOS is the first and only Disk Operating System to bring **STANDARD** Small Computer Systems Interface (SCSI) device drivers to the Color Computer system, running under Basic.

What does this mean to you?

First, SCSI is an accepted, industry standard interface system. This allows you to run one or more Hard Disks of any size on your Color Computer in Basic or under OS-9. But wait! There's **MORE!**

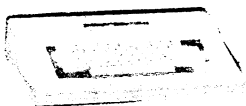
Because it provides generic SCSI drivers, RGB-DOS will **ALSO** run devices which you may not have **TODAY**, but will need to run **TOMORROW**, such as Streaming Tape Backups for the Hard Disk, CD ROMS and of course, Optical (Laser) Disk Drives!

RGB-DOS, combined with the **KEN-TON** or **DISTO** SCSI Interface, will assure you that your Color Computer System will be compatible with the ever growing line of mass storage products available both **TODAY AND TOMORROW.**

Here are some of the many features of RGB-DOS:

- Hundreds of systems in daily use - **PROVEN QUALITY**
- Runs **ALL** popular **COCO** software - **FULLY COMPATIBLE**
- Will **AUTO-BOOT ANY** program from Hard Disk or Floppy Disk
- Full Screen Directory Display for **COCO 1, 2 and 3**
- Automatic Drive Number, Free Space & Disk Name Display
- Supports Double Sided Floppy Disk Drives
- Reliable Floppy Disk Operation with the "High Speed Poke"
- Improved "COPY" command - Copy Filename to Drive Number
- "RUNM" Command for Machine Language Programs
- **FLEXIKEY** - Command Recall & Edit System by Colin J. Stearman
- Fixes the COPY bug, the DSKINI bug, the DOS bug & many others
- Will run two Hard Disk Drives of the same or different sizes
- Will run Streaming Tape Backup systems & Laser Disks
- Automatic Hard Disk Size detection. **NO** drivers to configure
- Works with the **LR-TECH, DISTO & KEN-TON** Hard Disk Interfaces
- **NO** extra RAM used or extra tokens added. **FULLY COMPATIBLE** with The Color Computer 1, 2 OR 3 **WITHOUT MODIFICATION**
- Easily Affordable. The RGB-DOS System Disk with Users Manual is only \$29.95 plus \$3.00 shipping & handling

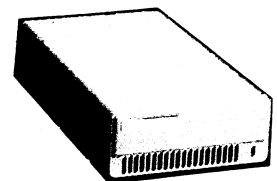
Don't wait! Supercharge your Color Computer **TODAY . . . with RGB-DOS!**



RGB COMPUTER
SYSTEMS

294 STILLWELL AVE.
KENMORE, NY 14217

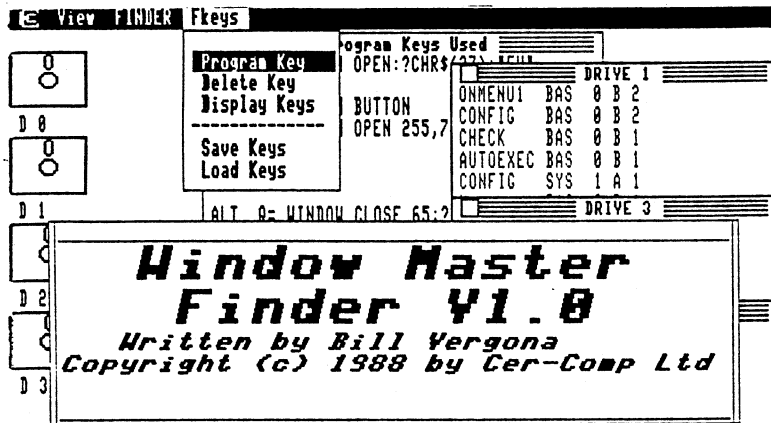
(716) 876-7538



"Window Master"

A Point & Click Window System
for the rest of us !!!

Fully Compatible with R.S. Dos
Enhanced Basic, it does not need
or use OS-9, and you don't have to
be a Rocket Scientist or a P.H.D.
to use Windows, Pull Down Menus,
Buttons, Icons, Edit fields or
Mouse Functions in your Programs!



Window Master Features

Multiple Windows

Window Master supports multiple window displays with up to a maximum of 31 windows on the screen. Overlapping windows are supported, and any window can be made active or brought to the top of the screen. Windows can be picked up and moved anywhere on the screen with the mouse. There are 6 different Window styles to choose from and the window text, border and background color is selectable.

Pull Down Menus

Menus are completely programmable with up to 16 menus available. They can be added or deleted at any time in a program. Menu items can be enabled, disabled, checked or cleared easily under program control. Menu selection is automatically handled by Window Master & all you have to do is read a function variable to find out which menu was selected.

Buttons, Icons & Edit Fields

Each Window can have up to 128 buttons, Icons or Edit fields active, if you can fit that many. Buttons, Icons and Edit field selection is handled automatically by Window Master when the mouse is clicked on one. All you have to do is read a Dialog function to find out which Button, Icon, or Edit field was selected, its very simple.

Mouse & Keyboard Functions

Window Master automatically handles the Mouse pointer movement, display and button clicks. It will tell you the current screen coordinate, the local window coordinate, window number the mouse is in, the number of times the button was pressed, which window number it was clicked in and more. The Keyboard is completely buffered, and supports up to 80 programmable Function keys that can contain any kind of information or command sequences you can imagine. You can load and save function key sets at any time. So, you can have special sets of function keys for different tasks. The "Ctrl" key is supported so that you have a full control code keyboard available.

Screen Display Fonts

Window Master supports up to 54 different character sizes on the screen with 5 different character styles. You can have Bold, Italic, Underlined, Super-Script, Sub-script or Plain character styles or any combination of them in any character size. You can also change the text color and background at any time to get really colorful displays.

Fully Basic Compatible

Window Master is fully compatible with Enhanced Color Disk basic with over 50 Commands & functions added to fully support the Point & Click Window System. Window Master does not take any memory away from Basic, so you still have all the Basic Program memory available.

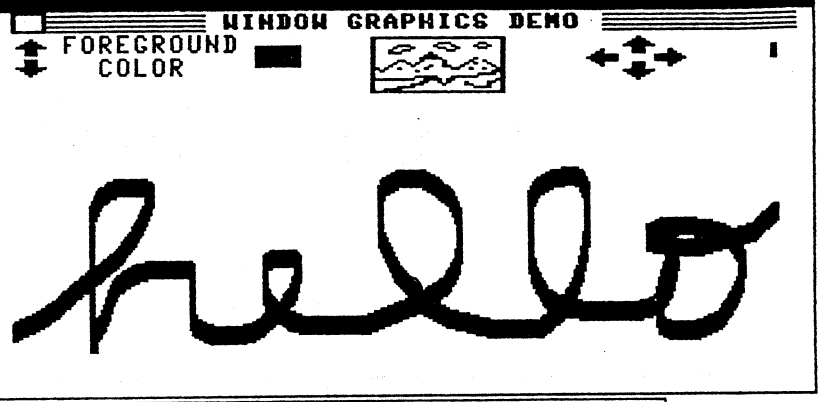
Hi-Resolution Displays

Window Master uses the full potential of the Color Computer 3 display by using the 225 vertical resolution display modes instead of the 192 or 200 resolution modes like most other programs. It uses either the 320/16 color mode or the 640/4 color display to give you the best display resolution possible, and can be switched to either mode at any time.

OPTIONS

```

CLEAR SCREEN 16,1,00,1,0,3,2,0
DOTS
BOX
CIRCLE
LINE
QUIT
-----
LOAD
SAVE
30 WINDOW OPEN 1,44,16,1,00,1,0,
3,2,0,"WINDOW GRAPHICS DEMO"
40 MW=1: MY WINDOW #=1
50 ON MENU GOSUB 540
70 MENU ON
80 PROTECT 3
90 ON DIALOG GOSUB 630: DIALOG 0
N
BREAK
OK
RUN
    
```



```

===== SAVE FILE =====
ENTER FILE TO SAVE
GFXTEST.PIC:2
    
```

Mixed Text & Graphics

Window Master fully supports both Text & Graphics displays and even has a Graphics Pen that can be used with HLINE, HCIRCLE, HSET and more. You can change the Pen width & depth and turn it on or off with simple commands. We also added Enhanced Graphics Attributes that allow graphics statements to use And, Or, Xor and Copy modes to display graphic information. With the Graphics enhancements added by Window Master, you could write a "COCOMAX" type program in Basic! In fact we provide a small graphics demo program written in Basic.

Event Processing

Window Master adds a powerful new programming feature to Basic that enables you to do "Real Time" Programming in Basic. It's called Event Trapping, and it allows a program to detect and respond to certain "events" as they occur. You can trap Dialog activity, Time passage, Menu Selections, Keyboard activity and Mouse Activity with simple On Gosub statements, and when the specified event occurs, program control is automatically routed to the event handling routine, just like a Basic Gosub. After servicing the event, the sub-routine executes a Return statement and the program resumes execution at the statement where the event occurred.

Enhanced Editing Features

Window Master adds an enhanced editor to Basic that allows you to see what you edit. It allows you to insert & delete by character or word, move left or right a word or character at a time, move to begin or end of line, toggle automatic insert on/off or just type over to replace characters. The editor can also recall the last line entered or edited with a single key stroke. You can even change the line number in line to copy it to a new location in the program.

Window Master Applications

Window Master pushes the Color Computer 3 far beyond its normal capabilities, into the world of a "User Friendly" operating environment. We are already planning several new programs for use with Window Master. So you don't have to worry about having to write all your own programs. And don't forget that many existing Basic and M.L. programs will run under Window Master with little or no changes. The Possibilities for Application programs are endless: Spread Sheets, Word Processing, Communications, Education, Games, Graphic Design, Desk Top Publishing and on and on.

Hardware Requirements

Window Master requires 512K of memory, at least 1 Disk Drive, a Hi-Res Joystick Interface and a Mouse or Joystick.

Technical Assistance

If you run into difficulty trying to use some of Window Master's features, we will be happy to assist you in any way possible. You can write to us at the address below or call us between 10am and 2pm Pacific Standard Time for a more timely response. Sorry, no collect calls will be accepted.

Ordering Information

To order WINDOW MASTER by mail, send check or money order for \$69.95, plus \$3.00 for shipping & handling to the address below. To order by VISA, MASTERCARD or COD call us at (702)-452-0632 (Monday thru Saturday, 8am to 5pm PST)

CER-COMP Ltd.
 5566 Ricochet Avenue
 Las Vegas, Nevada 89110
 (702)-452-0632

FILES DATE

		CALENDER V 2.0					
		May 1988					
		MON	TUE	WED	THU	FRI	SAT
New							
Open							
Save							
Save As...							
Init							
Quit							
	1	2	3	4	5	6	7
	8	9	10	11	12	13	14
	15	16	17	18	19	20	21
	22	23	24	25	26	27	28
	29	30	31				

The Chicago Rainbow Fest opens today, and we will be there with Window Master!

CoCo 'N Amateur Radio

Mike Dooley KE4PC

Editor's Note: In our last issue we left out part of the Basic program listing Mike had supplied to us. To our faithful CoCo 'N Amateur Radio column readers we offer our apology. We are running the entire listing in this issue. Mike's regular column follows this listing.

RTTY.BAS

```
5 REM THIS PROGRAM IS PUBLIC DOMAIN, BY N6LQV
10 CLEAR 200,&H4000:C=3
20 D=(PEEK(&HC000)=68):REM TAPE:
  D=0, DISK: D=-1
30 IF D THEN LOADM"RTTY" ELSE CL
  OADM"RTTY"
40 DEFUSR0=&HE00:DEFUSR1=&HE03
50 DEFUSR2=&HE06:DEFUSR3=&HE09
60 DEFUSR4=&HE0C:DEFUSR5=&HE0F
100 A=USR0(0):R=0:T=0
110 CLS:AUDIO ON:MOTOR OFF
120 PRINT"**** RTTY TERMINAL PROGRAM ****"
130 PRINT"
  BYTES"
140 PRINT"  SAVE RECEIVE BUFFER
  ";R
150 PRINT"  PRINT  ''  ''"
160 PRINT"  SHOW  ''  ''"
170 PRINT"  RECEIVE/TRANSMIT"
180 PRINT"  LOAD TRANSMIT BUFFER
  ";T
190 PRINT"  CLEAR ALL BUFFERS"
200 PRINT"  BAUD RATE: ";
210 ON B+1 GOTO 220,230,240,250
220 PRINT"45 WPM: 60":GOTO 260
230 PRINT"50 WPM: 67":GOTO 260
240 PRINT"57 WPM: 75":GOTO 260
250 PRINT"74 WPM: 100":GOTO 260
260 PRINT
270 PRINT"DURING RECEIVE/TRANSMIT:
  "
280 PRINT" BREAK: RETURN TO THIS
  MENU"
290 PRINT" CLEAR: TOGGLE RECEIVE
  /TRANSMIT"
300 PRINT" RIGHT ARROW: TRANSMIT
  BUFFER";
310 A=C*32:PRINT@A+32," ";:PRIN
  T@A+64,"->";:PRINT@A+96," ";
350 FOR I=0 TO 30:NEXT
360 A$=INKEY$
370 IF C>0 AND (PEEK(341) AND 8)
  =0 THEN C=C-1:GOTO 310
380 IF C<6 AND (PEEK(342) AND 8)
  =0 THEN C=C+1:GOTO 310
390 IF A$<>CHR$(13) THEN 360
400 ON C+1 GOTO 800,500,550,600,
  700,100,950
500 IF (PEEK(&HFF22)AND1) THEN 1
  10
24
```

```
510 A=USR5(-2):PRINT"  ":GOTO 1
  10
550 CLS:A=USR5(0)
560 IF INKEY$="" THEN 560 ELSE 1
  10
600 CLS:PRINT"RTTY RECEIVE  M
  ARK--><--SPACE":R=USR1(B):GOTO 1
  10
700 S$="LOAD FILE":GOSUB 900
710 IF A$="" AND D THEN 110
720 CLS:PRINT@229,"LOADING ";A$
730 IF D THEN F=1 ELSE F=-1
740 OPEN"1",F,A$:A=USR2(0)
750 IF EOF(F) THEN 790
760 LINEINPUT #F,A$
770 A$=A$+CHR$(13):A=USR3(A$)
780 IF A THEN 750
790 CLOSE F:T=USR4(0):R=0:GOTO 1
  10
800 IF R=0 THEN 110
810 S$="SAVE FILE":GOSUB 900:IF
  A$="" THEN 110
830 CLS:PRINT@229,"SAVING ";A$
840 IF D THEN F=1 ELSE F=-1:GOTO
  870
850 PRINT@293,"ARE YOU SURE (Y/N
  )":INPUT S$
860 IF S$<>"Y" THEN 110
870 PRINT@293,"":OPEN"0",F,A$
880 A=USR5(F):CLOSE F:GOTO 110
900 CLS:PRINTS$:PRINT
910 LINEINPUT"FILENAME?";A$
920 RETURN
950 B=B+1:IF B>3 THEN B=0
960 PRINT@269,"";:GOTO 210
```

MAKERTTY.BAS

```
10 REM THIS PROGRAM IS PUBLIC DO
  MAIN, BY N6LQV
20 PCLEAR4
30 CLS
40 PRINT"  RADIO TELETYPE TRANS
  CEIVER"
50 PRINTSTRING$(32,"-");
60 PRINT
70 PRINT"NOW GENERATING MACHINE
  LANGUAGE"
80 PRINT
90 PRINT"PLEASE WAIT..."
100 ST=&HE00:AD=ST:LI=900
110 READA$,CS
```

```
120 IF A$="X" THEN 200
130 FOR I=1 TO 64 STEP 2
140 A=VAL("&H"+MID$(A$,I,2))
150 POKE AD,A:CS=CS-A:AD=AD+1
160 NEXT
170 IF CS THEN PRINT"DATA ERROR
  IN LINE";LI:END
180 PRINT@174,949-LI
190 LI=LI+1:GOTO 110
200 IF PEEK(&HC000)=68 AND PEEK(
  &HC001)=75 THEN B$="DISK" ELSE B
  $="TAPE"
210 PRINT@96,"PROGRAM IS NOW IN
  MEMORY AND"
220 PRINT"READY TO BE SAVED. INS
  ERT ";B$
230 LINEINPUT"AND PRESS ENTER ";
  A$
240 IF B$="DISK" THEN 280
250 CSAVEM"RTTY",ST,AD-1,CS
260 LINEINPUT"PRESS ENTER TO SAV
  E AGAIN ";A$
270 GOTO 250
280 SAVEM"RTTY.BIN",ST,AD-1,CS
290 END
900 DATA16003216011516006816006C
  160032BDB3EDD76FBE4013BC40152418
  A680AD9F,2607
901 DATAA002966F26F1AD9FA00027EB
  AD9FA00027FA20E3398D2D8E7FFF9F17
  8E40299F,3863
902 DATA048D219E04BF1202BF12C89F
  069F08308904009C1723029E179F139F
  15DC0483,2591
903 DATA4029204E3404C64020033404
  5F1F9B35848E4029BF4004398DEAE684
  EE029E04,2792
904 DATA308904009C17242F108E0EBB
  A6C03404847FC6FE810D270C8020250E
  813F2502,2570
905 DATA8020E6A61F985D1703C63504
  5A26DDCCFF8BDB67EB4F44F5F20F7FF
  2D313429,3944
906 DATA803A2B2F3280802C233C3D36
  3733212A30352726382E3E8080803980
  43594E49,2235
907 DATA414D5A54464B4F525C4C5856
  574A4550475E535D555180808025800F
  6215D805,2685
908 DATA2102900DFC13DB04A902550C
  44116B0417020B09730D6A03260193BD
  B3ED8608,2365
909 DATA3D8E0EFB3AEC84FD1131EC02
  FD1158EC04FD1043FD1064EC06FD1092
  7FFF4017,3880
910 DATAFF228E03004F6F804A26FB0F
  02CCFEF8DD000F0A0F0B8E13739F0E9F
  260F220F,2820
```

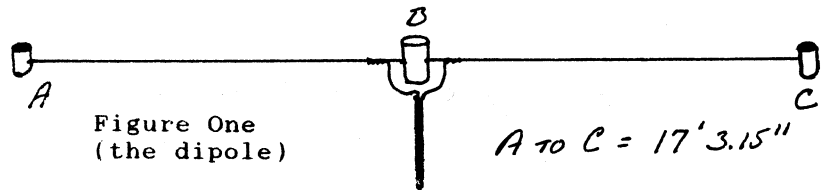

911 DATA230F1D8E0420CC6060ED818C
060025F98E05E09F0C86BFA7848E05A0
9F24869F, 3412
912 DATAA7841A50170087D62326048D
1620F51CAF17FED6AD9FA00026FAFC40
15B34013, 3367
913 DATA16FF0F9E26A6862B3A1F8981
602502C0408D50816026028D449E24A7
808C05C0, 3098
914 DATA251A8E0420EC8820ED818C05
A025F6CC6060ED818C05C025F98E05A0
9F24869F, 3779
915 DATAA784394C260CC60D8D198660
A79F402420D04C26058E139320064C26
058E1373, 2615
916 DATA9F26399E159C172404E7809F
1539962227031700F496232701390F1E
0F1F0F1B, 2156
917 DATAC6138D78DC1A8152250721FE
5A2A0A20DD5C2B022003CC007FD71B4F
D61AD31E, 2966
918 DATADD1E8300002406AC01AC0120
D3DD1E86059720CC0013971B971C8D40
4FD61AD3, 2901
919 DATA1EDD1E830000240FDC1A8152
C900D71B0C1C12C61220E3DD1ED61B58
D11C0621, 2752
920 DATA0A2026CFAC94C6088D124FD6
1AD31EDD1E83000025EE962144444439
8D4C4FD3, 3140
921 DATA1EDD1ECC00028D42CB028D3A
CB028D36D71AC059502B022003CC0000
C10F2202, 2628
922 DATA2003CC000FD11D26063D3DAC
8B20138E0410A68584BFA785961DD71D
E686CA40, 3157
923 DATAE78617012639AC01AC018601
5CC1602504A1012005B5FF2027F25CC1
602504A1, 2918
924 DATA012005B5FF2026F2398E6405
CE142BA6COA7808C040D25F7CC343CF7
FF21B7FF, 3646
925 DATA018602B7FF200F1E0F1F9E06
9C042724CC0000DD1186059720E6809F
06D7215F, 2727
926 DATA8D54AC94CC00085A26FD0421
8D480A2026F2EC9B3DAC8BCC0000DD11
538D3796, 3344
927 DATA226C7CC343CB7FF21F7FF01
8E0405CE1423A6COA7808C040D25F739
B6FF208A, 3731
928 DATA02B7FF20862A971039B6FF20
84FDB7FF20862797103924048DE22004
8DEB2000, 3444
929 DATA801D4A26FDB6FF2088FCB7FF
20D6104FD31EDD1E931124088D0F9610
802320E2, 3601
930 DATADD1E39AC943DAC0139960226
7CDC00C037498A0124022003CCFEF8DD
008E0300, 3057
931 DATA3AB7FF02B6FF008A801F89E8
84E48421FEA7848607DD029E049C0626
C69C0826, 3805
932 DATAC48E00009F069F089F043996
02263ADC00C037498A0124022003CCFE
F8DD008E, 2959
933 DATA03003AB7FF02B6FF008A801F
89E884E48426BEA78496014C27088B37
970121FE, 3530
934 DATA20818607DD0216FF7A4A9702
D601CB08D7010403247D867FB7FF02B6
FF008440, 3295
935 DATA27058E13B320058E13EB2000
A685E6852B289E04980AD70A84602609
C41FE780, 3004

936 DATA3D21FE2012C5402604861B20
04861F2000C41FED81A1019F04399E04
5C260E96, 2526
937 DATA0A8440970ACC0004E7803D20
EA5C260BCC0802A780ED81A18B20DC5C
2649D622, 3280
938 DATA270D8E40299F068E00009F04
9F08393DAC8B39D60B2747C004D70B8E
05E03AEC, 2844
939 DATA84ED88E0CC6060ED81EC84ED
88E0CC6060ED84D60B27043DAC84398E
05E09F0C, 4453
940 DATA86BFA78421FE395C26060322
AC843D395C26070F220323AC84393DAC
01399E08, 2605
941 DATA9C042607CC000C5A26FD39A6
809F089E0EA6862B129E0CA7808C0600
24139F0C, 2690
942 DATA86BFA784AC8B394C260F1286
60A79F400C8620970B3D3084394C2609
8E13939F, 2987

943 DATA0EA101200B4C26078E13739F
0E20013D3D12398045FF416053495580
44524A4E, 2303
944 DATA46434B545A4C57485950514F
4247FE4D5856FD8073FF6D605E787780
6474676C, 3436
945 DATA617A68756269726376707179
7F66FE6E6F7BFD8043594E49414D5A54
464B4F52, 3457
946 DATA5C4C5856574A4550475E535D
5551258080FDFDF363733212A30352726
382E3E2C, 2581
947 DATA233C3DFEFCFB808080808080
43594E49414D5A54464B4F525C4C5856
574A4550, 3262
948 DATA475E535D5551258080FDFDF80
2D313429803A2B2F32808080808039FE
FCFB8080, 3659
949 DATA808080524543454956456054
52414E534D49540000000000000000
00000000, 1621
950 DATAX, &HE00

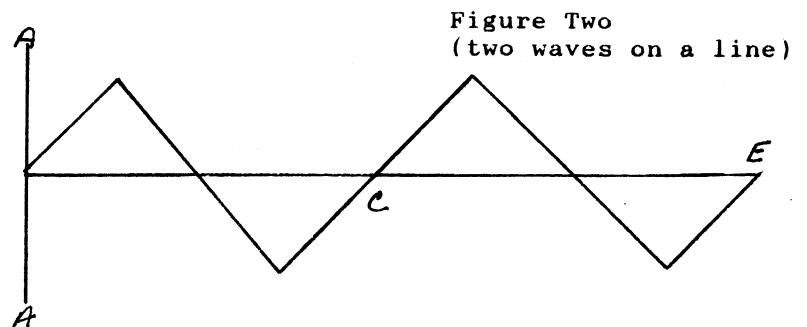
Hmmmm... CoCo's and Amateur Radio. What else can we do with this combination? How about antennas? Every Amateur needs a good antenna to operate a radio. Radios just don't work very well without them!

A good, simple, antenna is the dipole. Figure 1 shows what the dipole will look like. There are insulators at each end of the dipole and an insulator in the center where we'll attach the feedline.



A dipole is one half wavelength long which is the distance from point A to C. A wavelength depends on the frequency we're trying to operate at. Since different frequencies have different wavelengths we'll need to do two things in order to build our dipole. First, what frequency will we be operating at? And second, how long is a half wavelength at that frequency?

For the first item, let's build our antenna so it operates at 28.5 MHz. In order to discover the second item we'll need to start by learning what a wavelength is.



Every frequency has a corresponding wave length. Frequencies are measured in Hertz, or cycles per second. A wavelength is the length of wire it takes for one wave, or cycle, of a given frequency. Figure 2 shows two cycles, or waves, of a frequency. The first cycle is from point A to point C. The second is from point C to point E.

Since we're building a half wavelength antenna, we only need to know how long half a wavelength is. There's a formula that allows us to calculate a half wavelength if we know the frequency.

$$\text{Length (ft)} = 492/\text{frequency (MHz)}$$

The MHz in parenthesis stands for Mega Hertz. We already know that Hertz is cycles per second. Mega means million. A frequency of 144 MHz is a short way of saying 144,000,000 cycles per second.

The formula can be entered directly to calculate the length of wire needed for a half wavelength antenna, but it's more fun to use a short program (Figure Three).

```
10 CLS:W=0:F=0
20 INPUT "FREQUENCY (MHZ)";F
30 REM PRINT "A HALF WAVELENGTH
ANTENNAE WILL":PRINT"BE ";492/F;
" FEET LONG"
34 W=492/F:FT=INT(W):IN=(W-FT)*1
2
```

```
35 PRINT "A HALF WAVELENGTH ANTE
NNAE WILL":PRINT"BE ";FT;"FEET "
;IN;"INCHES"
40 PRINT "AGAIN?";
50 AS=INKEY$:IF AS="" THEN 50
60 IF AS="N" OR AS="n" THEN END
ELSE 10
```

Let's practice using the program by calculating the length of an antenna for a frequency of 10 MHz. I came up with 49 feet 2.4 inches. Your result should show the same.

Since we're going to build the dipole to operate at 28.5 MHz use the program and discover how long our antenna should be. According to my calculations our antenna should measure 17 feet 3.15 inches in length (I rounded it a little). That's the length from point A to C in Figure One.

Now that we know the length of our antenna, cut a piece of wire to that length. Attach insulators to each end of the wire. Find the center of the wire (point B in Figure One) and place another insulator there by cutting the wire. Connect the center conductor of the coax to one side of the dipole and the shield to the other side.

The coax I'm using is 50 ohm coax which can be purchased at Radio Shack (along with everything else here). The impedance of the center feedpoint where the coax is attached is 75 ohms. This means that, even under ideal conditions, we will always have a 1.5 to 1 mismatch (75 is 1.5 times 50).

Connect the antenna to your radio with an SWR bridge (meter) in line. You'll only need to do this SWR check if you will be transmitting. If you're just listening in (reading the mail) don't worry about the SWR. We're going to measure the Standing Wave Ratio of the antenna and 'prune' it to resonance. Even though we used the formula and program to tell us how long the wire should be for our antenna, it still must be checked and trimmed to get the best possible match. This is because the formula calculates an antenna's length in free space. The actual length of the antenna will be affected by its proximity to other conductive or semiconductive objects.

Measure the SWR and trim the antenna at the ends (points A and C) by removing about an inch at a time from the length. Be sure to remove an equal amount from both ends! After each trimming measure the SWR again until you reach 1.5 to 1. Once the antenna is trimmed it can be used on the air.

Well, our dipole is built and seems to be working pretty well. Next time we'll add to our program and learn how to build a three element beam! Now that's exciting! 73's de Mike Dooley KE4PC

Rustler: Keep the cowpoke from swinging from the gallows in this word guessing game. PMODE3 graphics designed to be attractive on both monochrome and color screens and good sound make this an enjoyable challenge. Includes editor program to create your own word lists. **\$5.00**

Blip: Test your memory and try to repeat the sequence of flashing squares and tones. Increasing levels of difficulty make this game a true challenge. **\$3.00**

Tippee: Try to get the best score in this familiar poker dice game. Up to nine players makes this game fun for the entire family. **\$3.00**

Special: All three games **\$8.00**

All games are in BASIC. 32K Extended Color Basic™ required.

Kcal Version 2.0: Use your DMP printer to create personalized calendars from 1988 on. Each day has three lines of 15 characters each available for your own personal notes. 99% machine language for fast response. 32K Extended Color Basic required. **\$6.00**

All prices are cassette or disk. Canadian money same as U.S. All others U.S. funds. Washington state residents add 7.5% sales tax. All shipping and handling included.

Send check or money order to: King Cottage Industries
1814 Valley St. NE
Poulsbo, WA 98370

Extended Color Basic is a registered trademark of Tandy Corp.

SEEC: To Change

Nancy Ewart

One of the first things I found out about computers way back in the early 60's was that computer people took the English language and put totally new meanings to common words, with the end result that as a non-computer person trying to talk to a programmer, I found myself listening to a language that I thought I knew but that had totally no meaning. Even now, learning the vocabulary is often the hardest problem to solve. I know this puts off many people who would like to learn more of the capabilities of computers. Herewith, some definitions for programmers when readings those erudite articles....

Pseudo code: sort of an outline in English of the different parts of the program and what you want them to do.

Source code: the program, in readable C, before it has been compiled.

Object code: the product of the compiler; the program translated into machine code. (Ah, the light dawns! The same source code run through different compilers produces machine specific code. That's how you get portability.)

Relocatable object code: Object code after the compiler has translated the source code into machine code but the memory addresses have not been absolutely defined; only the offset (or relative) information is kept. The most common example would be Library functions before linking.

Linker: the last step in compiling a C program; this is when the Library relocatable object code is added to the program. At this time the actual addresses are created for that program.

Function: These are C's building blocks. Every program must have one and only one main() function that starts the program. When you see source code without main(), you know that it is intended to be used within another program. You can use functions that are already defined in a Library; you can create your own library of functions that you use in programs over and over again; or you can create functions for a specific program.

```
EXAMPLE main()  
    getfacts();  
    sortfacts();
```

```
    printfacts();
```

A function must be defined someplace, at the end of the source code or included in a library.

Argument: a value that is passed to a function. EX getfacts(list); The value can be a single digit to a string. A function can have no arguments, (but you always include the parenthesis), or multiple arguments. EX add(a,b,c,d,e);

Library: There is a standard library that comes with the C Compiler. It can be changed and added to. This Library includes all the I/O functions like printf() in relocatable object code.

Compile time: what happens while you are running your source code through the compiler...usually errors, frequently fatal.

Run time: what happens when you are running your program after compilation ... sometimes unexpected.

Debug: when you change your source code to get rid of the errors.

(Then there is object-oriented programming that the slick magazines are talking about. This appears to be a way of defining classes of things hierarchically so that you propose actions related to the class before defining the unique object that will be affected. C++ is the language that handles data abstraction and object-oriented programming.)

SOURCES

All this, of course, is dependant upon a compiler. Microware's C Compiler for OS-9, published by Radio Shack can be obtained through Express Order at most Radio Shack stores for \$99.95. (Cat. # 26-3038 CMC) In addition, there are various shareware and public domain compiler offerings on Compu-Serve and the like.

A very helpful addition to the compiler is the C-Programmer's Tool Kit, #9 from the OS-9 User Group Library. The command cb is a pretty printer. If you type your source code

Clearbrook Software Group

NEWSLETTER

Version 2 of CSG IMS has been delayed but you should receive it by the time you get this newsletter.

MSF

The source code for MSF is now available for those who are developing MSDos disk drivers for other OS9 computers. The price is \$45.

Users of MSF (the MSDos file manager) are still encountering problems when accessing an MSDos disk as an OS9 disk. The most common mistake is when an MSDos disk is referenced by the OS9 drive name. For example: DIR /D1 instead of DIR /B (NOTE: there is a new version of MSDIR renamed to DIR which will work with both OS9 and MSDos disks). Another mistake is to leave a default directory on an OS9 disk and then replace it with an MSDos disk. When a command cannot be found or you forget to specify the drive, the default directory will be accessed.

As discussed in a previous newsletter, these problems occur because OS9 and MSDos use a different disk sector size. When an MSDos 512 byte sector is read into an OS9 256 byte buffer the buffer overflows into system memory. In the worst case this will cause a system crash but it could create other havoc as well.

The OS9 file manager which allocates the buffers for OS9 sectors (256 bytes) is RBF. If we patch RBF to allocate 512 byte sector buffers so system memory will be protected (except from Japanese MSDos disks which use 1024 byte sectors). More system space will be used for file buffers so you may not be able to run as many processes or windows or have as many files open. This example is for the following version of RBF.

```
OS9:ident rbf -m
Header for: RBF
Module size: $122E #4654
Module CRC: $EFBE13 (Good)
Hdr parity: $D4
Edition: $1C #28
Ty/La At/Rv: $D1 $81
File Man mod, 6809 obj, re-en, R/O
```

The BOOTPATCH command will patch the boot file and update the CRC. You should run BOOTPATCH on a backup of your system disk because if you make the wrong patch, your disk could become non-bootable.

Place your system disk in drive /d0 and your backup system disk in /d1. Make sure you have the BOOTPATCH program (you will have to enter and assemble the program listing below) in your CMDS directory. Enter the following commands.

```
OS9:bootpatch /d1/os9boot #40k
l rbf
c 2bb 01 02
c 73b 01 02
q
OS9:
```

Now boot the backup system disk so RBF will allocate 512 byte buffers for all files.

If you have a different version of RBF you can still make the patch if you have a disassembler or debugger such as ERINA. Simply find the two places in the code where there is a LDD #\$0100 followed closely by an OS9 F\$SR1MEM and an OS9 F\$SRQMEM. Use BOOTPATCH to change LDD #\$0100 to LDD #\$0200. NOTE: On the Smoke Signal VAR computer the OS9 calls in RBF may be JSR instructions.

```
*****
* Bootpatch.a - patch a module on disk
*****
* (c) Clearbrook Software Group
* Date: October 1988
* Author: Paul Kehler

ifpl
use ...../defs/os9defs
endc

Edition equ 1
LF equ 10 line feed
CR equ 13 carriage return
SP equ 32 space
BUFSIZE equ 256

mod modlen,modname,PRGRM+OBJECT,REENT+1,modstrt,memsiz

modname fcs 'bootpatch'
fcb edition

file rmb 1 path number
chflag rmb 1 NE means CRC needs update
rewrite rmb 1 NE means file needs update
offset rmb 2 change offset
oldval rmb 1 value to change
newval rmb 1 new value
module rmb 2 address of module in memory
filend rmb 2 pointer to end of file buffer
filesize rmb 2 size of file
buf rmb BUFSIZE
filebuf rmb 2000 actually to end of memory
stack rmb 200
memsiz equ .

modstrt cira the start of the program
clrb
stb
stb <chflag
stb <rewrite
modst0 ldb ,x+
cmpb #CR end of name?
lbeq modst2 ..yes, no options
cmpb #SP
beq modst0
cmpb #'-
beq modst1 ..check option
leax -1,x
tst <file
bne twonames ..two file names
lda #3 open for update
OS9 I$OPEN open the file
bcs error
sta <file
ldb #SS.Size
pshs x
pshs u,x
OS9 I$GETSTT get file size
bcs error ..report error
stu ,s
ldu 2,s
stx -2,s is file too large?
bne nomem ..yes
ldd ,s get file size
std <filesize
leax filebuf,u
stx <module default link to first module
addd <filesize
bcs nomem ..not enough memory
std <filend save pointer to end of file
addd #200
bcs nomem ..not enough memory
pshs d
cmps ,s++ is there room for the stack
blo nomem ..not enough memory
ldy ,s get file size
ldx <module get buffer pointer
lda <file
OS9 I$READ read the file
bcs error
ldd ,x
cmpd #$87cd is it module?
bne badfile
leas 4,s
```

```

puls x
bra modst0
nomem leax moremem,pcr
nomeml bsr writeln
clrb
OS9 F$EXIT
badfile leax notmod,pcr
bra nomeml

modst1 ldb ,x+
cmpb #CR
beq modst2
cmpb #SP
lbeq modst0
cmpb #'?
beq modsyn ..yes, report syntax
lbra modst0

twonames leax namerr,pcr
bsr writeln
modsyn bsr syntax show command syntax
clrb
error OS9 F$Exit

syntax pshs x,b
leax synmsg,pcr
bsr writeln
puls x,b,pcr

writeln ldy #synlen
lda #1 error path
OS9 I$Writeln
rts

skipspl ldb ,x+
cmpb #SP
beq skipspl
lbra toupper

skipspl bsr skipspl
leax -1,x
rts

modst2 tst <file was a file specified
bne modst25 ..yes
leax nomsg,pcr
bsr writeln
bra modsyn
modst25 leax buf,u point to buffer
ldy #BUFSIZE
clra
os9 I$READLN read a command
bcs error1
bsr skipspl skip leading spaces
cmpb #'L is it link?
lbeq link
cmpb #'C
beq change
cmpb #'Q
beq quit
cmpb #CR
beq modst25
leax unknown,pcr
modst28 lbsr writeln
bra modst25

error1 cmpb #E$EOF is it end of file?
beq quit ..yes, quit
OS9 F$EXIT

quit bsr changed check if module was changed
ldb <rewrite has any change been made?
beq quit9 ..no
pshs u
ldx #0
tfr x,u
lda <file
OS9 I$SEEK rewind to beginning of file
puls u
bcs quit9 ..error
leax filebuf,u point to data
ldy <filesize
OS9 I$WRITE write the file back
bcs quit9
clrb
quit9 OS9 F$EXIT

* update the CRC of changed module
changed pshs d,x,y,u
ldb hflag has module changed?
beq changed9 ..no
stb <rewrite file needs update
clr <chflag next module not changed yet
ldx <module
lda #8
clrb
changed1 eorb ,x+ find header parity
deca
bne changed1
comb
stb ,x update header parity
ldx <module
ldd 2,x get module size
subd #3
leau d,x point to crc
tfr d,y module size less accumulator
ldb #8
stb ,u

```

```

stb 1,u
stb 2,u
OS9 F$CRC calculate the new CRC
com ,u
com 1,u
com 2,u
changed9 puls d,x,y,u,pc

change bsr gethex get offset
lbcx modst25
std <offset
bsr gethex get old value
lbcx modst25
stb <oldval
bsr gethex get new value
lbcx modst25
stb <newval
ldx <module point to module
ldd <offset
cmpd 2,x
blo changel
leax outofrng,pcr .. offset out of range
lbra modst28
changel leax d,x point to byte to change
ldb <oldval
cmpb ,x
beq change2
leax diffrent,pcr
lbra modst28 ..print warning
change2 ldb <newval
stb ,x
ldb #1
stb <chflag set flag
lbra modst25

gethex clr ,-s
clr ,-s
lbsr skipspl
cmpb #CR
beq gethex25 ..no number
gethex21 cmpb #CR
beq gethex29 ..number complete
cmpb #SP
beq gethex29 ..number complete
cmpb #'0
blo gethex25 ..error
cmpb #'9
bls gethex22 ..0-9
andb #8 convert to upper case
cmpb #'A
blo gethex25 ..error
cmpb #'F
bhi gethex25 ..error
subb #'A-'0-10
gethex22 subb #'0

lsl 1,s
rol ,s
lsl 1,s
rol ,s
lsl 1,s
rol ,s
lsl 1,s
rol ,s
orb 1,s
stb 1,s
ldb ,x+ get next character
bra gethex21
gethex25 leax hexmsg,pcr
lbsr writeln
comb set carry
gethex29 puls d,pcr return hex value

link lbsr changed check if module changed
lbsr skipspl
pshs x
link0 ldb ,x
cmpb #SP
bls link01
bsr toupper convert name to upper case
stb ,x+
bra link0
link01 ldb -1,x
orb #80
stb -1,x set hi bit on last byte
leax filebuf,u
pshs d,x
ldd ,x
cmpd #87cd is it proper module?
bne link99 ..no
link1 stx ,s save pointer to current module
tfr x,d
addd 2,x get module size
std 2,s point to next module
ldd 4,x

```

```

leax d,x point to current module name
ldy 4,s link name
link2 ldb ,x+
bsr toupper
cmpb ,y+
bne link3
tstb
bpl link2
ldx ,s ..module found
stx <module
leas 6,s
lbra modst25 go for next command
link3 ldx 2,s
cmpx <filend is it end of file?
blo link1 try next module
link99 leax notfound,pcr
leas 6,s fix stack
lbra modst28 and go report error

* convert B to upper case
* bit 7 not affected
toupper pshs b
andb #87f mask bit 7
cmpb #'a
blo toupper1
cmpb #'z
bhi toupper1
andb #5f convert to upper case
toupper1 orb #80 set hi bit
andb ,s+
rts

unknown fcc "Unknown command!"
fcb CR
moremem fcc "Not enough memory to load
file!"
fcb CR
namerr fcc "Only one file allowed!"
fcb CR
nomsg fcc "No file specified!"
fcb CR
hexmsg fcc "Non hex character encountered!"
fcb CR
outofrng fcc "Offset is not within module!"
fcb CR
diffrent fcc "*** old value not the same,
not patched ***"
fcb CR
notfound fcc "Module not found!"
fcb CR
notmod fcc "File does not contain
module(s)!"
fcb CR

synmsg
fcc "Syntax: bootpatch [opts] <filename>
[opts]"
fcb LF
fcc " where opts are: -? for help"
fcb LF
fcc " <filename> is the name of the file
containing the module to"
fcb LF
fcc " be patched. NOTE: this is
different from MODPATCH"
fcb LF,LF
fcc " Commands: L <module> link to the
specified module in the file"
fcb LF
fcc " C <offset> <old value>
<new value> change the old value"
fcb LF
fcc " at offset to new
value."
fcb LF
fcc " Q quit"
fcb LF,LF
fcc " The CRC of the module will be
automatically updated when you link"
fcb LF
fcc " to a new module or quit."
fcb LF,CR

synlen equ *-synmsg

emod

modlen equ *

end

```

Clearbrook Software Group

BOX 8000-499
Sumas, EA
U.S.A. 98295-8000
(604)853-9118

BOX 8000-499
Abbotsford, B.C.
CANADA V2S 6H1


```

001:/* ZMORT1/CCC 09/11/88 */
002:/* This is a DEVELOPMENTAL version */
003:#include stdio/h
004:#include math/h /* Required for atof() */
005:main(ac,av)
006:    int ac;
007:    char *av[];
008:
009:    float in = 0;
010:    float pay = 0;
011:    float loan = 0;
012:    float ipd = 0;
013:    float ppd = 0;
014:    int i = 1;
015:
016:    long tod;
017:    char buf1[81];
018:    char tbuf[9];
019:
020:    FILE *ptr,*fopen();
021:
022:/* initialization of tbuf and &tod */
023:sysdate(tbuf);
024:time(&tod);
025:
026:pay = atof(av[1]);
027:in = atof(av[2]);
028:loan = atof(av[3]);
029:
030:/* Information to be on screen */
031:puts("Written by Walt Gabriel, 20 Wren Place, Toms River, N.J., 08753");
032:puts(" 201 240-5084");
033:
034:strcpy(buf1,(__FILE__)); /*dis@pays file name */
035:strcat(buf1,"Compiled on   ");
036:strcat(buf1,(__DATE__));
037:puts(buf1);
038:
039:if (ac == 4) /* Note count is 4 - 0,1,2,3 */
040:
041:    if ((ptr = fopen("*pr","w+")) == NULL)
042:        perror("fopen error");    exit(-1);
043:    fputs("MORTGAGE PAYOUT TABLE",ptr);
044:    fputs(" \n\n",ptr);
045:    fprintf(ptr,"Original Loan Value    %8.2f\n",loan);
046:    fprintf(ptr,"Annual Interest        %6.2f\n",in);
047:    fprintf(ptr,"Monthly Payment        %6.2f\n\n",pay);
048:
049:    fputs("PAYMENT    PRINCIPAL    PRINCIPAL    INTEREST \n",ptr);
050:    fputs("NUMBER      REMAINING    PAID        PAID        \n\n",ptr);
051:
052:while (loan >=0)
053:
054:
055:/* computation */
056:    ipd = loan * in/1200;
057:    ppd = pay - ipd;
058:    loan = loan - ppd;
059:
060:fprintf(ptr," %d    %8.2f    %6.2f    %6.2f \n",i++,loan,ppd,ipd);
061:
062:fputs(ctime(&tod),ptr); /*prints date, time */
063:fclose(ptr);
064:
065:else
066:    puts("DATA ENTRY FORMAT\n");
067:    puts("DMORTPAY PAYMENT/MONTH ANNUAL INTEREST LOAN\n");
068:    puts("EXAMPLE: MORTPAY 132.98    8.5    5000 \n");
069:

```

070:
071:

Here are the changes that I made so that the source code would compile on my CoCo.

```
On line 003 #include <stdio.h>
On line 004 #include <math.h>
double atof(); /*between lines 13 & 14 */
```

Delete lines 16, 17, 18

```
pffinit(); /* insert on line 21 */
```

Delete lines 22, 23, 24

Delete lines 34, 35, 36, 37

```
On line 41 change "*pr" to "/p"
On line 42 change perror to ferror
```

Delete line 62



6 - 2 - 1

\$83.25

SAVE 15% FROM LIST PRICE

Just in time for holiday giving it's 6-2-1! "6-2-1" is our new program to save your money! Six back issues of *Clipboard*, a two-year subscription to *Clipboard* and one year of *ClipDisk*. Normally \$97.97, now you can save 15%, only \$83.25! Just write in "6-2-1" on the order form on page 18.

MC - Visa - Discover accepted.

And now you can phone in your order!
We accept VISA, MasterCard and Discover - (716) 679-0126 — 9 a.m. to 9 p.m. Monday thru Saturday!



SPECIAL DEAL ON 500 PROGRAMS IS BACK!

BACK BY POPULAR DEMAND! GET OUR LATEST 50 DISKS OR TAPES FULL OF OVER 500 PROGRAMS. HERE IS WHAT YOU'LL RECEIVE:

- ★ Over 250 Utility/Home Application Programs including a Word Processor, Database, Spreadsheet, Disk Utilities, Business Software, Electronics Series, Educational Programs for Kids, plus much more!
- ★ Over 200 exciting games including King Pede, Kron, Star Trek, Flight Simulator, Wizard, Horse Races, Football, plus much more.
- ★ Over 30 adventures including Rambo, Haunted House, Power Sword, Skid Row, plus 32k graphic adventures.

Individual issues sell for \$9⁰⁰ each or \$450⁰⁰ for all 50. We slashed the price to only \$150⁰⁰!

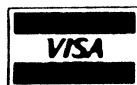
REG. \$450



\$150⁰⁰

★★THIS MONTH ONLY★★

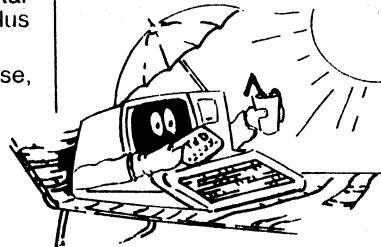
Buy this package of 500 programs and receive a free 6 month subscription.



COOL DOWN TO OUR SUMMER PRICES ON SUBSCRIPTION SOFTWARE

THERE IS NO BETTER WAY TO ENJOY THE SUMMER THAN TREATING YOUR COLOR COMPUTER TO 10 READY-TO-RUN PROGRAMS EACH MONTH. GET 12 DISKS OR TAPES A YEAR CONTAINING OVER 120 QUALITY PROGRAMS. A SUBSCRIPTION TO T & D SOFTWARE CONSISTS OF 10 READY-TO-LOAD PROGRAMS DELIVERED BY FIRST CLASS MAIL EVERY MONTH.

NO, WE ARE NOT THE SAME AS THE RAINBOW ON TAPE. IN FACT, MANY SUBSCRIBERS HAVE WRITTEN IN AND SAID THAT WE ARE MUCH BETTER THAN RAINBOW ON TAPE!



PRICES

	TAPE	THIS MONTH ONLY
1 YEAR (12 issues)	20.00	60.00
6 MO. (6 issues)	40.00	35.00
1 ISSUE	9.00	8.00

Michigan Residents Add 4%
Overseas Add \$10 to Subscription Price
Personal Checks Welcome!

- ★ Available on COCO 1.2 and 3
- ★ Includes Documentation
- ★ Over 4,500 Satisfied Customers
- ★ Back Issues Available From July '82 (Over 720 Programs)



PLEASE SPECIFY TAPE OR DISK

T & D SUBSCRIPTION SOFTWARE, 2490 MILES STANDISH DR., HOLLAND, MI 49424 (616) 399-9648

T&D Subscription Software, an advertiser in this magazine, wants to introduce you to their variety of inexpensive software by giving you a:

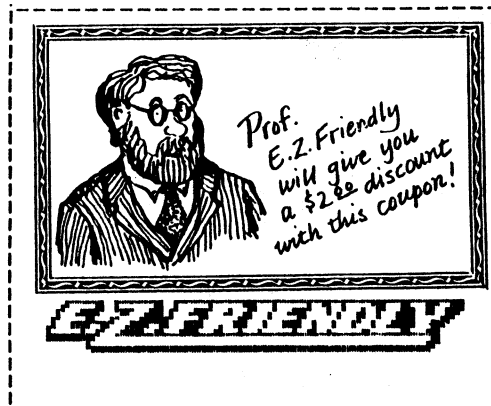
FREE Disk or Tape Filled with 10 programs

1 coupon per household

Circle one: TAPE DISK

Mail this coupon to:
T&D Free Offer
 P.O. Box 1256
 Holland, MI 49423

WITH THE COUPON
\$5.00 OFF
ANY COUPON & YOUR OLD ADDRESS
\$15.00 OFF
TRK & TRK3 DOS
 6000 BOREATION
 5003 B ST.
 PHILA. PA 19120-3024
 315-457-1009



Save 10% on C.S.G. with this coupon
Clearbrook Software Group



Information Management System

CSG IMS is a full featured relational database manager for the Color Computer and OS9.

- Interactive access to databases for quick queries.
- Includes a structured compiled language supporting program modules with full parameter passing and recursion.
- User defined screen and report formats.
- Record, index, and file sizes are virtually unlimited.
- Text, BCD floating point (14 digits), short and long integer, and date data types.

CSG IMS for CoCo2/3 OS9 L1/2 (single user) \$169.95
 CSG IMS for OS9 L2 or 68000 (multi user) \$495.00
 CSG IMS demo with manual \$30.00
 OTHER CSG PRODUCTS:
 ERNA Symbolic User-mode Debugger for OS9 \$69.00
 SERINA System-mode Debugger for OS9 Level 2 \$139.00
 MSF MS-DOS File Manager for CoCo3/OS9 L2 \$45.00
 MSF with SDSK3 \$65.00
 Shipping - N. America: \$5, Overseas: \$10

Clearbrook Software Group, Inc.
 U.S.: P.O. Box 8000-499, Sumas, WA 98295
 CANADA: P.O. Box 8000-499, Abbotsford, B.C. V2S 6H1
 Phone: (604)853-9118



OS9 is a trademark of Microsoft Systems Corp., MS-DOS is a trademark of Microsoft Corp.

STOCK MARKET PORTFOLIO FOR THE COLOR COMPUTER

Save 10% on Stock Market Portfolio with this Clipboard Coupon

Paparis Enterprises
 700 York St.
 Williamsburg, VA 23185

MARTY'S RUBBER STAMP DISCOUNT COUPON!

We can now offer COLORS for many nylon cartridge ribbons. Our RAINBOW-PAK contain 6 different colored ribbons. Write for prices. Specify your printer's name & model.

Marty's Rubber Stamp Shop
 47 South Second St.
 SAINT CLAIR, PA 17970
 (717) 429-1052

Buy BOTH HELLO.BAS and GOOD GAMES TRIO from R.C. PIERCE Software and receive a free sample copy of CoCo Clipboard OR we'll extend your subscription an extra issue!

P.O. BOX 1787,
 Main Post Office,
 Edmonton, AB, Canada
 T5J-2P2
RCPIERCE SOFTWARE
 PH: (403) 474-8435

SAVE \$15 WITH THIS COUPON ON A SCSI INTERFACE WITH R.T.C. OR \$15 ON A 2 CHANNEL DUAL COMM BOARD

FROM:

KEN-TON ELECTRONICS
 187 GREEN ACRES RD.
 TONAWANDA, NY 14150

◀◀ GIMMESOFT ▶▶

This certificate entitles

to 10% (TEN) off

their next purchase.

Non-transferable and not valid with any other sale prices or certificates.

This offer expires: Sept 1, 1989

CoCo Clipboard

KING COTTAGE INDUSTRIES

Include this coupon with any order before Christmas and receive one free blank diskette

Save \$5.00 on

R.A.D. PRODUCTS

Textform

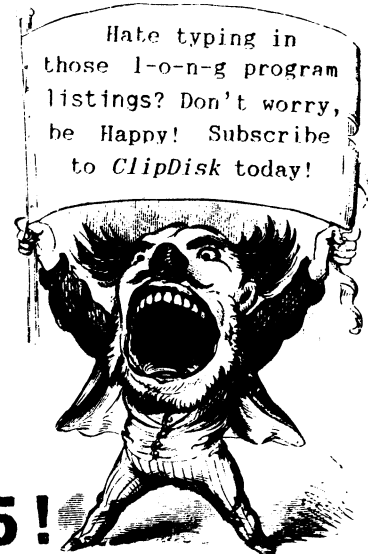
Exp. 12/31/88

CCMDisk Service

PRESENTS

ClipDisk

FULL YEAR (6 issues) Only \$49.95!



Now all the BASIC, C, BASIC09, Assembly Language and OS9 programs you see in *CoCo Clipboard Magazine* are available on disk. With *ClipDisk* you'll save time and be more productive computing with your Color Computer.

A subscription to *ClipDisk* is \$49.95 for 6 issues. Single issues are available as well for just \$9.95. Of course in order to get the full benefit of *ClipDisk* you'll need the corresponding issue of *CoCo Clipboard Magazine*. All *ClipDisks* are mailed first class.

So don't delay, get your subscription in today!

I've enclosed (check one) () 1 year of *ClipDisk* \$49.95
 () single copy 9.95

Specify month desired _____

Canadian customers can also obtain *ClipDisk* for \$49.95 when paid in U.S. Funds.

Name: _____
 Address: _____
 City: _____
 State/Prov: _____
 Zip/Postal Code: _____

6 - 2 - 1
 \$83.25
 SAVE 15% FROM LIST PRICE
 Just in time for holiday giving it's 6-2-1! "6-2-1" is our new program to save your money! 6 back issues of *Clipboard*, a 2 year subscription to *Clipboard* and 1 year of *ClipDisk*. Normally \$97.95 now you can save 15% only \$83.25! Just write in "6-2-1" on the order form on page 18.
 MC-Visa-Discover accepted.

Please charge my purchase to my (circle) VISA / MasterCard / Discover Card.

MasterCard Interbank Number

Card # _____
 Exp. Date _____
 Signature _____



PLEASE NOTE: All charge card orders are handled courtesy of On-Line Computer Services. Your charge card billing will show On-Line Computers and NOT CoCo Clipboard Magazine.

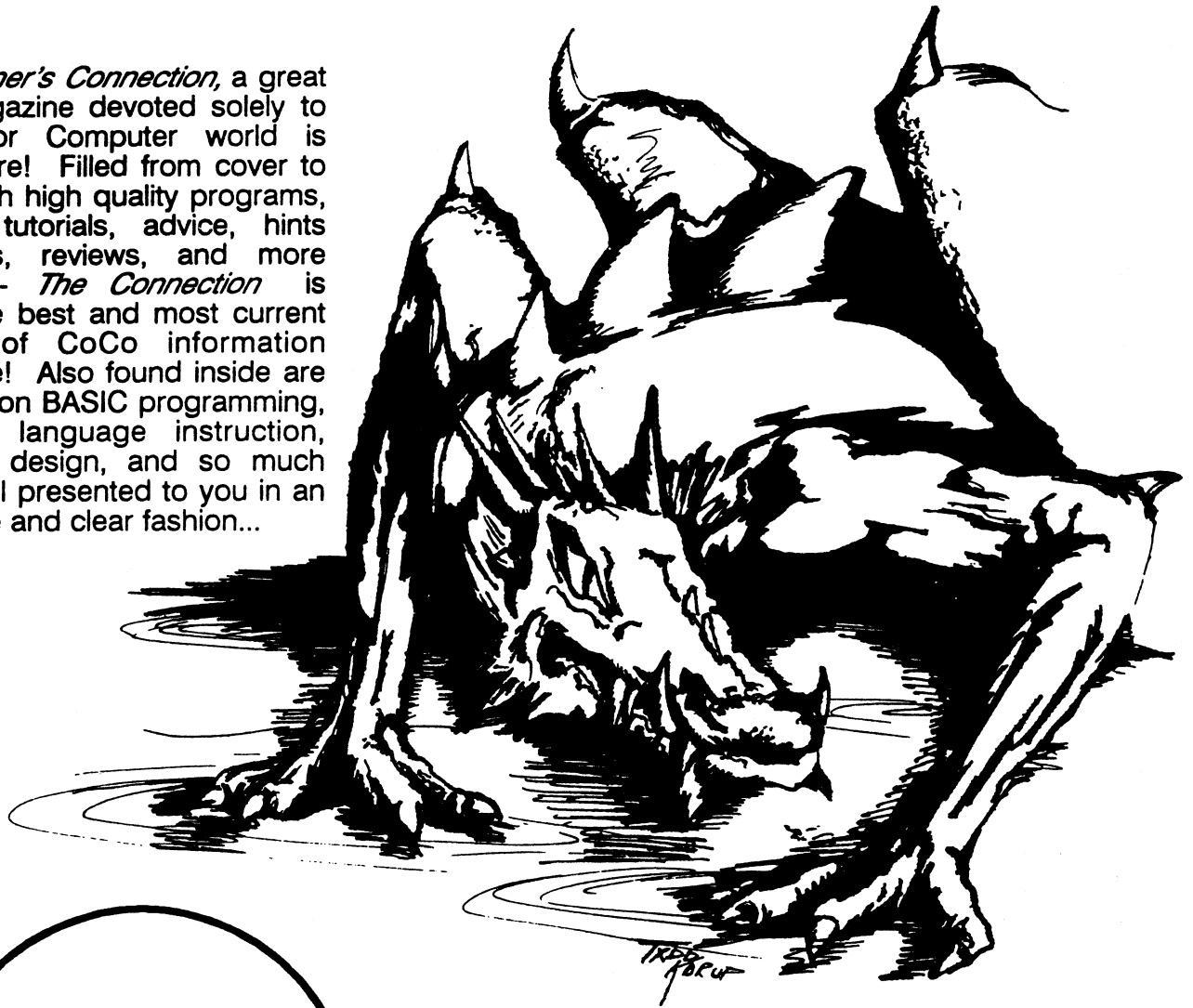
CoCo Clipboard Magazine

3742 U.S. 20, Box 3, Fredonia, NY 14063

The Gamer's Connection!

The ONLY magazine devoted solely to the CoCo gaming world!

The Gamer's Connection, a great new magazine devoted solely to the Color Computer world is finally here! Filled from cover to cover with high quality programs, articles, tutorials, advice, hints and tips, reviews, and more reviews - *The Connection* is truly the best and most current source of CoCo information anywhere! Also found inside are columns on BASIC programming, machine language instruction, program design, and so much more! All presented to you in an enjoyable and clear fashion...



Subscribe Now
for only

\$15.00

Right now you can take advantage of our introductory special and save on our subscription rates. You can receive a full year (six issues) of *The Gamer's Connection* for only \$15.00 (\$18.00 Canadian residents (US funds)). This special offer is in effect only for a very limited time, so act now and start your subscription to *The Gamer's Connection* today!

Please send your check or money order to:

The Gamer's Connection - Dept. C, P.O. Box 102, Grafton, WI 53024

Real BASIC for OS9!

Only \$39.95

```
OS-9 LEVEL TWO VR. 02.00.01
COPYRIGHT 1986 BY
MICROWARE SYSTEMS CORP.
LICENSED TO TANDY CORP.
ALL RIGHTS RESERVED

July 11, 1988 14:37:30

Shell

OS9: xmode /w6 type=0
OS9: lniz /w6
OS9: rdb <=>/w6 &
&007
```



```
RSB COPR. 1988 BURKE & BURKE
DISK EXTENDED COLOR BASIC 2.1
COPR. 1982, 1986 BY TANDY
UNDER LICENSE FROM MICROSOFT
AND MICROWARE SYSTEMS CORP.

OK
LOAD "DEMO"
OK
LIST
10 PMODE 4:SCREEN 1,1
20 X=RND(256)-1:Y=RND(192)-1
30 XS=RND(256-X)-1:YS=RND(192-Y)-1
40 LINE (X,Y):(X,XS,Y,YS),PSET,BF
```



Welcome to ...

R. S. B.

See R.S.B. and other Burke & Burke products at the Princeton Rainbowfest. Don't miss our Hard Disk Seminar!

There is nothing wrong with your Color Computer. Do not attempt to adjust it. The BASIC you know and love is now running under Level 2 OS9 windows. You are in command.

Burke & Burke is proud to present another OS9 programming language: Disk Extended Color BASIC.

You've probably heard of this language. It's the one your Color Computer was born with. We're talking PMODE, DIR, COLOR, RENUM, PLAY and other familiar words. Under Level 2 OS9. In as many windows as your memory lets you create.

Our R.S.B. software creates a complete, OS9-compatible version of Disk Extended Color BASIC by reading your CoCo's ROM chips. Burke & Burke has added new software to give you OS9-style graphics, sound, printer, and disk I/O. Of course, you can't use R.S.B. to run machine language programs, and some BASIC commands work slightly differently under R.S.B.

R.S.B. loads and saves files using OS9's file format, so we've also included utilities to transfer BASIC programs and data files between OS9 and BASIC disks. Your BASIC programs can take advantage of great OS9 features like hard disks, no-halt floppies, multi-tasking, and 2 MHz operation.

You must have a CoCo 3 with at least 128K RAM, and a floppy controller with Disk Extended Color BASIC 1.0, 1.1, 2.0, or 2.1 ROM, or CoCo 3 CDOS ROM, to use R.S.B.

Works with
all hard disk
systems!



Runs in 128K;
512K
preferred.

R.S.B. \$39.95

Hard Disk Interfaces

We've sold hundreds of our affordable, high-performance hard disk interfaces to Color Computer enthusiasts worldwide!

Each includes an anodized housing, 100 page user manual, and software for use with OS9. The CoCo XT-RTC adds a battery-powered real time clock / calendar for OS9 and BASIC.

CoCo XT \$69.95
CoCo XT-RTC \$99.95

XT-ROM: Install XT-ROM in your hard disk controller's BIOS ROM socket. It automatically boots and reboots OS9 from your hard disk.
\$19.95

HYPER-I/O

Now BASIC runs hard drives, big floppies, and more!

HYPER-I/O modifies the RS-DOS BASIC in your CoCo 1, 2, or 3 to provide a "Dynamic Disk Interface". Use your existing BASIC and RS-DOS software with hard disk interfaces (CoCo XT, DISTO, LR), RAM Disks, and any mix of floppy drives from 160K to 720K each.

Fully RESET protected, user configurable, expandable, EPROM-able HYPER-I/O V2.6

\$29.95

HYPER-III (Adds RAM Disk and Print Spooler for CoCo 3)
\$12.95

HYPER-I/O is now compatible with DISTO, LR, and other SASI Hard Disk Systems!

One Year of Burke & Burke!

Burke & Burke has been offering our innovative line of Color Computer products to the public for one year now. Our customers often thank us for the fine service and support that we have given them. Now it's our turn to say, "THANK YOU!"

What started out as a dream has become reality at Burke & Burke because of you and the hundreds of customers that have purchased our products in 1988. We thank you, above all, for the priceless recommendations and endorsements that many of you have given other CoCo users.

Our sincere thanks, too, go out to the magazine publishers, SIG / BBS operators, club officers, and Color Computer dealers who have provided us with the means to promote, demonstrate, or support our products.

Thank You!
and Season's Greetings

CoCo XT Answers to some common questions about the CoCo XT.

- A true "NO HALT" hard disk system
- Controls 1 or 2 hard drives
- Full ECC and CRC error correction
- Average access 30% faster than SASI
- Uses PC-type hard disk controllers & drives
- Does not use or disable interrupts
- Compatible with most RS-232 interfaces
- 20 Meg system cost: under \$450
- Requires Multi-PAK or "electric" Y-cable

Buy a drive, Western Digital controller, and case from the PC dealer of your choice. Plug them into the CoCo XT, plug the CoCo XT into your Multi-PAK, and install the OS9 or BASIC software. Presto!

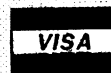
OS9 Utilities

Wild & MV Version 2.1

Use "wildcards" with most OS9 commands, or rearrange your directory tree. Features recursive directory searches. A hard disk must! **\$19.95**

EZGen Version 1.04

Powerful OS9 bootfile editor. Change module names, add or delete modules, patch bytes, or rearrange modules. Works on other files, too. **\$19.95**



Burke & Burke

P.O. Box 1283 Palatine, IL 60078-1283 (312) 397-2898

ILLINOIS RESIDENTS PLEASE ADD 7% SALES TAX. COD's add \$2.20. Shipping (within the USA) \$2.00 per CoCo XT; \$1.50 per disk or ROM. Please allow 2 weeks for delivery (overnight delivery also available for in-stock items). Telephone orders accepted (312) 397-2898.

Multi-Vue Icon Editor

Randy Krippner

Multi-Vue is a fascinating program. It has drawbacks, of course, but it is still one of the most interesting programs to come along for the Coco 3. One of Multi-Vue's features is that it permits a program to be run by simply clicking on an icon.

When you do a directory of an OS9 disk under Multi-Vue by clicking on one of the disk drive icons, Multi-Vue displays the files on the disk, together with an icon for each file.

Normally these icons are the standard Multi-Vue default icons. But other icons can be created by the user for the purpose of identifying programs and making them into "point and click" executable programs.

Making a program runnable from the Multi-Vue directory display by clicking on an icon is easy. You need two things; an icon and an Application Information File, or AIF.

The AIF is a special file to Multi-Vue. When MV does a directory of a disk, it looks for files which are preceded by the characters "aif.". These files give Multi-Vue the information it needs in order to properly handle a particular program or data file. It contains information that tells Multi-Vue the name (pathlist) of the program or file, how much memory it requires, what type of window the program needs, how large the window must be, etc. It also contains the pathlist of the icon to be displayed. When Multi-Vue does a directory of a disk, it does not list the actual AIFs on the disk. Instead it displays the icon specified in the AIF above the name of the program listed in the AIF. It's all very simple, really. Unfortunately, Tandy and Microware complicated things by failing to provide a way to create new icons.

There are icon editors available in the public domain on CompuServe, but not everyone has access to them. So IconEdit was born.

IconEdit is a program that will let you draw your own icons for Multi-Vue. It will also create new AIFs to make programs into "auto-run" programs that can be executed under Multi-Vue by double clicking on an icon. IconEdit requires Multi-Vue (of course), the hi-res mouse interface, and 512K RAM. It MUST run on a 4 color, 320 X 192 screen. If you try to use IE in the 16 color mode or in one of the 640 modes, I don't know what will happen, but the resulting icon will probably bear no resemblance to what you drew originally.

So before you start to type in IE, you need an AIF to start up Basic09 on a 320 X 192, 4 color screen. The following listing will do just that. It's an AIF which will display the Demo icon above BASIC09. When you double click on it, Basic09 will start in the correct graphics mode and with more than enough memory to use Icon Edit.

Basic09

```
icons/icon.demo
85
8
40
24
0
2
```

You can use BUILD or EDIT to create this AIF. Once you have IconEdit running, you can use it to make new AIFs without having to bother with a text editor.

Save this as "aif.b94", for Basic09, 4 color mode, or use some other name that will tell you what it is later. When you do a directory of this disk with Multi-Vue, just click twice on the Demo icon to start Basic09. (If you have more than one AIF using the Demo icon to start Basic09, you won't know which one is which, so put this one on a separate disk or separate directory.) Of course before this will work Basic09 has to be either loaded into memory already or in your current execution directory.

As was the case with Deluxe PowerGraph, you should have Basic09 merged with GFX2, Syscall and Inkey. I described how to do this for DPG, so I won't repeat it here.

IconEdit also requires that ERROR be in the execution directory or in memory, and that the ERRMSG file be in the SYS directory in order to handle the error trapping functions.

If you have DPG you can save time by re-using some of the procedures from it for IE. One of Basic09's more powerful features is the ability to write independent procedures which can be used in other programs.

If you have DPG, load it into Basic09 and save the following individual procedures from it:

```
ReadMouse
QMouse
```

DuArrow
 YesNo
 Oops
 WnSet
 GetFile
 Convert
 Delay

After saving these, empty the Basic09 work space by typing: kill*. Then load these procedures back into Basic09 and type in the rest of the procedures. Then save the whole thing to disk as IconEdit.

Using IconEdit is simple. After starting Basic09 in the 4 color, 320 X 192 graphics mode as described above, and loading Icon Edit, start it by typing "run main". The main screen will appear almost immediately.

The large area occupying the left half of the screen is the icon editing area. To the upper right is a menu with four options: Save Icon, Make AIF, Clear Screen and End Program. Beneath this menu is a small box which will display the icon in actual size.

The edit area shows the icon at a greatly enlarged size to make creating an icon easier. The edit area is made up of a box 24 pixels high and 24 pixels wide. Each "pixel" in the edit area is actually an 8 X 8 pixel block.

Drawing an icon is primitive but effective. To set a pixel, just put the pointer at the correct location in the edit area and click B1. To blank a pixel, put the pointer on the pixel, and click B1 again. Since an

icon is only 24 X 24 pixels, elaborate drawing functions are not necessary.

As you draw your icon, the image will appear at the same time, in actual size, in the small box to the right of the editing area so you can see what it will actually look like when Multi-View displays it.

There are no separate "modes" in IE. To select one of the functions on the menu, just point to it and click B1. You may need to hold the button down for a second because response time on the menu is slow. The function being selected will be highlighted as you move the pointer over it.

Save Icon will save your icon to disk when you're done drawing it. When prompted, type in the full pathlist for the icon. For example, if you wished to call your icon "phone", and wanted to store it in the ICON directory which, in turn, is in the CMDS directory, you would enter:

/d0/cmds/icons/phone <enter>

If a file by the same name already exists, you will be asked if it's alright to re-write it. Click on the YES box to go ahead and re-write it. Click on the NO box to abort the operation.

Make AIF will generate an Applications Information File for you. All you have to do is respond to the questions as indicated below:

Program name: Enter the name of the program you want executed. For example, if you want this AIF to start up Basic09, you would respond by entering BASIC09 or the full pathlist of the program.

Parameters: Some programs may require parameters to be entered when the program is started up. If this is necessary, enter these parameters here.

Icon file: The pathlist of the icon to be displayed for this program. If you wished to use the phone icon used in the example above, you would enter: "/d0/cmds/icons/phone".

Memory: The memory requirements of the program. Multi-View expects the amount of memory to be given in "pages", each 256 bytes long. But dealing with "pages" of memory is a bit difficult for most people, so IE expects you to enter the memory requirements in bytes. IE will convert the number of bytes into pages for you.

Screen type: The type of screen display required by the program. This is the "format code" of the required window as listed in the table on page 9-34 of the Basic09 reference manual that came with OS9 L2. The screen type for the 320 X 192, 4 color mode, for example, would be 6.

Window width: The width of the window in characters.

Window length: The height of the window in characters.

Alpha Software Technologies

OS9 Level II BBS Release 2.0

The best BBS system has just gotten better! System comes complete and ready to run in less than 5 minutes! Use the built in menus or create your own, you can even run your own programs or games on line! Complete message system allows easy message posting and retrieving. Complete file transfer system supports Xmodem and Ymodem as well as keyword searching single line and paragraph file descriptions. System runs completely in the background, allowing you full use of your computer! Also comes with it's own Terminal program "Quikterm" Free!

612K OS9 Level II Required.....\$29.95

The Zapper

This wonderful utility allows you to patch anything! Patch commands directly on the disk and fix CRCs automatically! Even allows you to patch the OS9boot file without making a new boot disk! Save files that have been lost or deleted! Fix crashed disks! Hundreds of uses!

64k OS9 Level I or II required.....\$19.95



Disk Manager Tree

This versatile utility will make your OS9 life a breeze! No more will you have to fight with complicated directory structures. No more searching for files and typing long path names. All of this is displayed using windows. A tree window allows you to change, create, and delete directories quickly. A files window allows you to copy, view and delete files easily. Perfect for the OS9 beginner! Multi-View compatibility makes it perfect for Multi-View users!

612K OS9 Level II Required.....\$29.95



Level II Tools

Finally OS9 life becomes easy! With these great utilities you'll be using OS9 like a pro! Complete wildcard commands make file manipulation simple and easy! Tree commands make directory and creating and maintaining windows a snap! Many other useful utilities make your OS9 life more pleasant! 26 great utilities for only \$24.95, that's less than \$1.00 per utility!

128k OS9 Level II Required.....\$24.95



Warn One

Finally a complete OS9 Level II windowing terminal that you can afford! The program's many features include Auto-dial, Auto macro, File transfers, buffer capture, on-line timer, chat mode and much, much more! Menu and dialog windows make it super simple to use! All you'll ever need in a terminal program! Comes complete with ICON for Multi-View compatibility!

612K OS9 Level II Required.....\$34.95



Multi-Menu

Easily create your own pop-down menus with this great utility! No programming experience necessary! With this utility you can run any OS9 command or program from a menu. Menu creation is super simple and super easy! Actually see the menu as it develops. A must for any Multi-View user!

612K OS9 Level II and Multi-View required.....\$19.95



Send check of money order to: Alpha Software Technologies
 P.O. Box 16522
 Hattiesburg MS. 39402

Or call: (601) 266-2773

Please add \$3.00 Shipping and handling. C.O.D. Orders add an additional \$2.00



Background color: The color of the background.

Foreground color: The color you wish the foreground (text) to be.

The other menu functions are:

Clear Screen: Will erase the screen and re-display the menu.

End Program: Ends the program and returns to the Basic09 prompt.

As usual, IconEdit can be obtained on the ClipDisk for this month if you don't want to type it in.

Deluxe Power Graph Note: DPG will not work properly if PACKed. The drawing functions will work, but the disk I/O functions will not. The problem seems to be related to how OS9 allocates memory for PACKed Basic09 programs. It appears that insufficient memory is being allocated to permit the graphics buffers to be mapped into the user address space. But since my schedule is pretty hectic at the moment, and since DPG works fine unPACKed, I'm not going to look into the matter further at this point in time. If you come up with a solution, let me know and I'll print it in a future column.

Questions and comments can be sent to me at:

Randy Krippner
1014 W. Hwy. 114, Lot 29
Hilbert, WI.

Note that the Unicorn BBS is temporarily out of service. When it comes back on-line, it will be with a new phone number. I'll publish it as soon as the board is operational again.

More than a year has passed and now...

He's Coming Back !!

It's Rush Caley & "Downtime!"

Is Rush the "Morton Downey, Jr." of the CoCo World? Read his comments on Rapid Fire Topics, Remedial Life, Quotable Quotes, and much more.

Rush is back and with a vengeance. Alternating with Downtime will be a new Q&A column on business applications. Start 1989 with a new or renewed subscription to Clipboard and Downtime. (Acid proof gloves not included)

While They Last Tandy 64K Color Computer II with Extended Color BASIC Only \$49.95

We just purchased 500 units directly from Radio Shack. These are new units, in original boxes with BASIC manual, switch box, and video cable.

Ideal Christmas gift, backup computer, or upgrade from a CoCo 16K machine!

Add \$5.00 for S&H. We will only accept orders that can be shipped by UPS to the continental 48 states. Discounts are available on large orders. VISA/MC, checks, school purchase orders, no CODs. NY residents add sales tax. 30 Day money back guarantee.

**Zebra Systems, Inc. 78-06 Jamaica Ave
Woodhaven, NY 11421 (718) 296-2385**

MJK & MJK3 DOS



WHY BUY ADOS 3
WHEN YOU CAN HAVE THIS

Now: MJK-DOS for COCO 1, 2, and 3 \$39.95
Most powerful operating system for the CoCo ever!
Allows up to 3 DS-80 track drives or standard drives. The DS-80 drives are software configurable to standard Radio Shack* format in order to maintain compatibility. Allows global file name specification with wildcards. All files will be displayed alphabetically, including the date that the file was saved. Use one command to KILL or COPY a number of files that meet the global filename specification (can be a full disk) in one run or one at a time upon user prompt (Y-key). Use the powerful CHAIN command to use programs of any length. Use the built in FULL SCREEN EDITOR to allow fast and easy program modification. You can even use the IRON command that will put you (after an error) in the modified line editor or get the automatic error trap routine with fully spelled out error names. Hit one key to repeat the last command. You will also get error trap, repeat key, AUTO, DATE, CAT (two columns of directory with only the filenames & extensions), WAIT, RUNM, BAUD, FIND, OLD, DATES, (string in basic program), LCOPY (groups of basic lines), REPL (to replace a string), IYPE (list a text file on screen/printer), SPLIT or JOIN basic lines, SAY for real spoken text, word peek & poke and many more... MJK-DOS is primarily intended for double-sided 80-track drives (720K each)
ALLOWS YOU TO READ/WRITE/FORMAT 35/40 DISKS ON A 80T DRIVE
***** EPROMABLE ***** FREE UPDATES FOR 1 YEAR *****

MJK512 DOS(COC03-512K)	\$49.95
BUILT IN RAM DISK AND RAM TEST COMMANDS	
Monitor-Disassembler (COCO 1, 2&3)	\$39.95
Source-Code Generator/LabelGenerator (COC01,2&3)	\$49.95
JB REMOTE rs-232 pack driver for bbs etc. (COC01,2,&3)	\$19.95
NEVKEY (COC03) new key scan--gives you true ALT & CTRL	\$15.00
NEVKEY232(COC03)--JB REMOTE and NEVKEY in one package	\$25.00
RTC -real time hardware clock for the coco 1,2,&3	\$35.00

CALL OR WRITE (COD ORDERS OK)
COCO CONNECTION OF PHILA. PA.
5005 B ST.

PHILA. PA. 19120
PHONE: 215-457-1809 VOICE AND DATA
COMPUSEKIVE 1D: 72317.437(LEAVE PHONE#)

```

0066  RUN readmouse(x,y,b1,b2)
0067  RUN gfx2("owend")
0094  END
PROCEDURE WnSet
0000  PARAM wtype:INTEGER
0007  TYPE registers=cc,a,b,dp:BYTE; x,y,u:INTEGER
002C  DIM regs:registers
0035  regs.a=0 \regs.b=$86
004C  regs.y=wtype \regs.x=0
0063  RUN syscall($E,regs)
0071  END
PROCEDURE GetFile
0000  PARAM t:STRING[6]; file:STRING[30]
0017  RUN gfx2("owset",1,0,4,38,4,2,0)
0039  RUN wnset(4)
0041  RUN gfx2("curon")
004E  SHELL "tmode echo"
005C  RUN gfx2("curxy",1,1)
006F  PRINT t;
0075  RUN gfx2("curxy",6,1)
0088  INPUT file
008D  SHELL "tmode -echo"
009C  RUN gfx2("curoff")
00AA  RUN gfx2("owend")
00B7  END
PROCEDURE Convert
0000  PARAM x,y:INTEGER
000B  x=INT(x/8)
0018  y=INT(y/8)
0025  END
PROCEDURE SetMouse
0000  TYPE registers=cc,a,b,dp:BYTE; x,y,u:INTEGER
0025  DIM regs:registers
002E  regs.a=0 \regs.b=$94 \regs.x=$0101
0051  regs.y=$FFFF
005D  RUN syscall($E,regs)
006B  regs.a=0 \regs.b=$89 \regs.x=$0301
008E  RUN syscall($E,regs)
009C  SHELL "tmode -echo"
00AB  RUN gfx2("scalesw","off")
00C0  RUN gfx2("curoff")
00CE  RUN duarrow
00D2  END
PROCEDURE main
0000  BASE 0
0002  DIM icon(24,24):INTEGER
0012  DIM b1,b2:BYTE
001D  DIM x,y,func:INTEGER
002C  DIM done,yes:BOOLEAN
0037  done=FALSE
003D  FOR x=0 TO 23
004D     FOR y=0 TO 23
005D         icon(x,y)=0
006B     NEXT y
0076     NEXT x
0081     RUN gfx2("clear")
008E     RUN setmouse
0092     RUN makebuff
0096     RUN menu
009A     REPEAT
009C         RUN qmouse(x,y,b1,b2)
00B5         RUN converL(x,y)
00C4         IF x<24 THEN
00D0             RUN EditIcon(icon)
00DA             ENDIF
00DC             IF x>27 AND y<5 THEN
00EF                 RUN select(func)
00F9                 ON func GOSUB 10,20,30,40
0110             ENDIF
0112             UNTIL done

```

```

PROCEDURE ReadMouse
0000  PARAM xv,yv:INTEGER; bt1,bt2:BYTE
0015  TYPE rt=jl:STRING[8]; b1,b2:BYTE; j2:STRING[18]; xval,yval:
INTEGER; j3:STRING[8]
004F  TYPE registers=cc,a,b,dp:BYTE; x,y,u:INTEGER
0074  DIM regs:registers; rat:rt
0085  REPEAT
0087     regs.b=$89 \regs.a=0
009E     regs.y=1 \regs.x=ADDR(rat)
00B7     RUN syscall($D,regs)
00C5     UNTIL rat.b1<>0 OR rat.b2<>0
00DD     xv=rat.xval \yv=rat.yval
00F3     bt1=rat.b1 \bt2=rat.b2
0109     END
PROCEDURE QMouse
0000  PARAM rx,yv:INTEGER; ba,bb:BYTE
0015  TYPE rt=jl:STRING[8]; b1,b2:BYTE; j2:STRING[18]; xval,yval:
INTEGER; j3:STRING[8]
004F  TYPE rg=cc,a,b,dp:BYTE; x,y,u:INTEGER
0074  DIM rat:rt; regs:rg
0085  regs.b=$89 \regs.x=ADDR(rat)
009F  regs.a=0 \regs.y=1
00B5  RUN syscall($D,regs)
00C3  ba=rat.b1 \bb=rat.b2
00D9  rx=rat.xval \ry=rat.yval
00EF  END
PROCEDURE DuArrow
0000  RUN gfx2("gset",202,1)
0013  END
PROCEDURE YesNo
0000  PARAM pmp:STRING[13]; yes:BOOLEAN
0012  DIM x,y:INTEGER; b1,b2:BYTE
0027  RUN gfx2("owset",1,11,4,15,7,2,0)
0049  RUN wnset(4)
0051  RUN gfx2("box",15,31,23,39)
0068  RUN gfx2("box",63,31,71,39)
007F  RUN gfx2("curxy",1,1)
0092  PRINT pmp;
0098  RUN gfx2("curxy",4,4)
00AB  PRINT "YES";
00B3  RUN gfx2("curxy",10,4)
00C6  PRINT "NO";
00CD  REPEAT
00CF     REPEAT
00D1         RUN readmouse(x,y,b1,b2)
00EA         UNTIL b1<>0
00F5         RUN convert(x,y)
0104         IF x=2 AND y=4 THEN
0117             yes=TRUE
011D             RUN gfx2("fill",16,32)
012F             x=999
0137             ELSE
013B                 IF x=8 AND y=4 THEN
014E                     yes=FALSE
0154                     RUN gfx2("fill",64,32)
0166                     x=999
016E                     ENDIF
0170                     ENDIF
0172                     UNTIL x=999
017E                     RUN gfx2("owend")
018B                     END
PROCEDURE oops
0000  PARAM enum:INTEGER
0007  DIM x,y:INTEGER; b1,b2:BYTE
001C  RUN gfx2("owset",1,5,8,30,6,0,2)
003E  RUN wnset(4)
0046  PRINT
0048  SHELL "error "+STR$(enum)
0057  PRINT
0059  PRINT "Click to continue"

```



```

011A END
011C 10 RUN SvIcon \ RETURN
0125 20 RUN MakeAIF \ RETURN
012E 30 RUN cls \ RETURN
0137 40 RUN YesNo("Really End?", yes)
0152 IF .yes THEN
015B RUN cur
015F END
0161 ENDIF
0163 RETURN
PROCEDURE EditIcon
0000 BASE 0
0002 PARAM icon(24,24):INTEGER
0012 DIM x,y,t:INTEGER
0021 DIM b1,b2:BYTE
002C DIM done:BOOLEAN
0033 done=FALSE
0039 REPEAT
003B RUN gmouse(x,y,b1,b2)
0054 IF x<191 AND b1<>0 OR b2<>0 THEN
006E RUN convert(x,y)
007D IF icon(x,y)=1 THEN
008F RUN gfx2("put",2,1,x*8,y*8)
00AE icon(x,y)=0
00BC t=0
00C3 ELSE
00C7 RUN gfx2("put",2,2,x*8,y*8)
00E6 icon(x,y)=1
00F4 t=1
00FB ENDIF
00FD RUN UpdateIcon(x,y,t)
0111 RUN delay
0115 ENDIF
0117 IF x>191 THEN
0123 done=TRUE
0129 ENDIF
012B UNTIL done
0133 END
PROCEDURE UpdateIcon
0000 PARAM x,y,t:INTEGER
000F RUN gfx2("color",t)
0021 x=x+247
002C y=y+103
0037 RUN gfx2("point",x,y)
004E RUN gfx2("color",2)
005E END
PROCEDURE Select
0000 PARAM func:INTEGER
0007 DIM selection(4):STRING[13]
0018 DIM x,y,t,loc(4):INTEGER
0030 DIM b1,b2:BYTE
003B DIM done:BOOLEAN
0042 done=FALSE
0048 FOR t=1 TO 4
0058 READ selection(t),loc(t)
0069 NEXT t
0074 REPEAT
0076 RUN gmouse(x,y,b1,b2)
008F RUN convert(x,y)
009E IF x<27 OR x>38 OR y<1 OR y>4 THEN
00BF GOSUB 100
00C3 func=0
00CA END
00CC ENDIF
00CE GOSUB 200
00D2 UNTIL done
00DA GOSUB 100
00DE END
00E0 100 FOR t=1 TO 4
00F3 RUN gfx2("curxy",27,loc(t))

```

```

010B PRINT selection(t);
0114 NEXT t
011F RETURN
0121 200 GOSUB 100
0128 RUN gfx2("revon")
0135 t=y
013D func=y
0145 RUN gfx2("curxy",27,loc(t))
015D PRINT selection(t);
0166 RUN gfx2("revooff")
0174 IF b1<>0 OR b2<>0 THEN
0187 done=TRUE
018D ENDIF
018F RETURN
0191 DATA "Save Icon",1,"Make AIF",2,"Clear Screen",3,"End Program",4
PROCEDURE menu
0000 DIM x:INTEGER; t:STRING[13]
0012 RUN gfx2("clear")
001F RUN gfx2("bar",192,0,194,191)
0036 RUN gfx2("box",246,102,272,128)
004E RUN gfx2("box",214,7,312,40)
0066 RUN gfx2("box",212,5,314,42)
007E FOR x=1 TO 4
008E READ t
0093 RUN gfx2("curxy",27,x)
00A8 PRINT t;
00AE NEXT x
00B9 END
00BB DATA "Save Icon", "Make AIF", "Clear Screen", "End Program"
PROCEDURE makebuff
0000 RUN gfx2("get",2,1,0,0,8,8)
001D RUN gfx2("bar",0,0,8,8)
0034 RUN gfx2("get",2,2,0,0,8,8)
0051 END
PROCEDURE cur
0000 RUN gfx2("curon")
000D SHELL "tmode echo"
001B RUN gfx2("killbuff",2,1)
0031 RUN gfx2("killbuff",2,2)
0047 END
PROCEDURE cls
0000 DIM yes:BOOLEAN
0007 RUN yesno("Clear Screen",yes)
0020 IF .yes THEN
0029 RUN menu
002D ENDIF
002F END
PROCEDURE MakeAIF
0000 DIM icon,aif,pgm:STRING[30]
0014 DIM mem,screen,col,row,bckgnd,frgnd:INTEGER
002F DIM prms:STRING[30]
003B DIM yes:BOOLEAN
0042 DIM path,enum:INTEGER
004D RUN gfx2("owset",1,1,3,37,16,0,2)
006F RUN wset(4)
0077 RUN gfx2("curon")
0084 SHELL "tmode echo"
0092 INPUT "Program Name: ",pgm
00A8 INPUT "Parameters: ",prms
00BC INPUT "Icon file: ",icon
00CF INPUT "Memory: ",mem
00DF mem=INT(mem/256)
00ED INPUT "Screen type: ",screen
0102 INPUT "Window Width: ",col
0118 INPUT "Window Length: ",row
012F INPUT "Background color: ",bckgnd
0149 INPUT "Foreground color: ",frgnd
0163 RUN getfile("AIF",aif)
0173 ON ERROR GOTO 100

```

Product Reviews

Review Crew

Dino Data Base
RAM Electronics
814 Josephine
Monmouth, Oregon, 97361

Pgm. Type: Educational
Requires : 128K Coco 3, disk
Price : \$29.95

Dino Data Base is a unique and fascinating program. While it is a data base program in that it has the ability to retrieve and display or print the information contained in the data base, that term is somewhat misleading. This program is more of an information retrieval system, since all of the information is already included with the program. The information contained in the data base is, of course, about dinosaurs.

The program permits you to search for a particular type of dinosaur by simply typing in the name of the dinosaur, or the first few letters of the dinosaur's name. (As an alternative, you can move forward or backward through the data sequentially by selecting the NEXT or BACK options from a pull down menu.)

After the dinosaur has been found, the main display screen will appear. This screen is split into four windows. The first shows the dinosaur's name, the pronunciation of the name, the name's Latin meaning, the Order and Family, the period of time in which it lived, and places where it was discovered. A second window displays a list of names of other dinosaurs that were related to the one selected. The third window displays a drawing of the dinosaur itself. The fourth window displays a world map, with indicators showing where the fossil remains of the dinosaur were found.

After the dinosaur is displayed, the menu will pop up, permitting you to move backwards or forwards through the file to view information about other dinosaurs, perform another search, end the program, or jump into another mode which permits the information about a dinosaur to be printed out on a printer.

The information about each dinosaur is limited, of course. It would be impossible to put everything known about these fascinating creatures into a single data file. But it is still quite interesting to flip through the data base, getting an idea of what these animals looked like, where they were found, etc.

This program is probably best suited for

grade school aged children, who seem absolutely fascinated by it. They would flip through the various dinosaurs that appear in the data base and use the search functions to find particular dinosaurs they were interested in.

Dino Data Base is a very well crafted program. The only problem I had with it was a very minor one; there is no option to change the printer baud rate within the program. If you wish to use a baud rate other than the default of 600, you will have to poke the appropriate value in before running the program.

The Memory Game
RAM Electronics
814 Josephine
Monmouth, Oregon, 97361

Pgm. Type: Game
Requires : 128K Coco 3, disk, joystick
Price : \$19.95

The Memory Game is a matching game similar to the old Concentration television show. Pairs of cards are placed at random, face down on the screen. The player uses the joystick to select two cards. These cards are turned over, displaying the symbol on the cards. If the symbols match the player gets credit for the match and may continue. If the cards do not match they are turned back over and the player has to try to make another match.

Up to four people may play the game, taking turns at attempting to find matches. If more than one person plays, the winner is determined by who makes the most matches. If one person plays, the player works against a timer.

Operation of the program is simple. Except when entering in the names of the players, the game is operated entirely by the joystick. The author ought to be congratulated on the excellent use of the Coco 3's 16 color hi-res graphics. The display is colorful, attractive and very well designed.

While the concept behind the game is simple enough so that even a pre-school age child can understand it and enjoy it, it is also challenging enough for adults. (My five year old son, beats me with embarrassing regularity.)

```

0179 10  CREATE #path,aif:WRITE
0188      PRINT #path,pgm
0192      PRINT #path,prms
019C      PRINT #path,icon
01A6      PRINT #path,mem
01B0      PRINT #path,screen
01BA      PRINT #path,col
01C4      PRINT #path,row
01CE      PRINT #path,bckgnd
01D8      PRINT #path,frgnd
01E2      CLOSE #path
01E8 20  RUN gfx2("owend")
01F8      RUN gfx2("curoff")
0206      SHELL "tmode -echo"
0215      END
0217 100  enum=ERR
0220      IF enum=218 THEN
022C          RUN yesno("Re-Write?",yes)
0242          IF yes THEN
024B              SHELL "del "+aif
0257              GOTO 10
025B          ELSE
025F              GOTO 20
0263          ENDIF
0265      ENDIF
0267      RUN oops(enum)
0271      GOTO 20
PROCEDURE Svlcon
0000      TYPE registers=cc,a,b,dp:BYTE; x,y,u:INTEGER
0025      DIM regs:registers
002E      DIM enum,path:INTEGER
0039      DIM yes:BOOLEAN
0040      DIM filename:STRING[30]
004C      RUN getfile("Name",filename)
005D      IF LEN(filename)<2 THEN
006A          END
006C      ENDIF
006E      ON ERROR GOTO 100
0074 10  CREATE #path,filename:WRITE
0083      RUN gfx2("get",19,1,248,103,23,24)
00A0      regs.a=1 \regs.b=$84
00B7      regs.x=$1301 \regs.y=1
00CE      RUN syscall($8E,regs)
00DC      regs.a=path
00E8      RUN syscall($8A,regs)
00F6      regs.a=1 \regs.b=$84
010D      regs.x=$1301 \regs.y=0
0124      RUN syscall($8E,regs)
0132      CLOSE #path
0138      SHELL "attr "+filename+" e pe"
014D      RUN gfx2("killbuff",19,1)
0163      END
0165 100  enum=ERR
016E      IF enum=218 THEN
017A          RUN yesno("Re-Write?",yes)
0190          IF yes THEN
0199              SHELL "del "+filename
01A5              GOTO 10
01A9          ELSE
01AD              END
01AF          ENDIF
01B1      ENDIF
01B3      RUN oops(enum)
01BD      END
PROCEDURE delay
0000      DIM x:REAL
0007      FOR x=1 TO 70
0019      NEXT x
0024      END
PROCEDURE kbuff
0000      RUN gfx2("killbuff",19,1)

```

The Memory Game is an excellent alternative to the mindless shoot-'em-ups that dominate the computer game market. The graphics are excellent, it's easy to play, but not easy to win. This is one of the few games on the market that the whole family will find entertaining. Highly recommended.

FKEYS III
Gimmesoft
P.O Box 421
Perry Hall, MD 21236

Pgm. Type: Function key utility
Requires : 64K Coco I/II/III, 1 drive
Price : \$19.95

I do a lot of work with my Coco. I'm at the keyboard for at least four or five hours a day or more, so I am constantly looking for software to make life with a computer a little easier. FKEYS III has proven itself to be one of the best function key utilities I've seen.

The basic idea behind function keys is this: Instead of having to type in a long string of characters to make the computer perform a task, function keys let you do the same task by simply pressing one or two keys instead of ten or twenty. Instead of typing something like "CLS:DIRO:?FREE(0) <ENTER>" all I have to do with FKEYS III loaded is press F1 on my Coco 3 keyboard.

FKEYS III comes with twenty predefined functions that can be activated by simply holding down the CTRL key, then pressing the function key. (The down arrow key is used as a control key on keyboards that lack the extra keys of the Coco 3 keyboard.)

Any of these functions can be easily redefined to fit your own needs. Just about any RS-DOS or Extended Color BASIC command can be programmed into a function key. The only restriction is that the function can only be up to 25 characters in length, which should be sufficient for almost any application.

FKEYS III can make life much, much easier. You can load and run programs, call up disk directories, do POKES or perform any function you would normally type in, just by pressing a function key.

And, of course, FKEYS III lets you save your redefined function keys to disk so you don't have to reprogram them every time you want to use the utility.

If this was all FKEYS III did, it would still be a very useful program. But there's much more.

FKEYS III lets you modify your DOS to handle 35 or 40 track disk drives, single or double sided drives, and can even change the step rates to increase the speed of the drive. It also resets the drive head to track 0 to prevent the annoying head banging problem that occasionally occurs if the drive head is in the wrong place when the drive is turned on.

There is even an option to save the modified DOS to a disk file which can then be used to burn an EPROM which can be used to replace the DOS ROM chip in your disk drive controller so that the FKEYS III functions you've defined can be available as soon as

Continued On 46

Master Basic09

Bill Brady

I just got home from the RainbowFest in Princeton where Ted reminded me that I am due for another column. Time sure flies!

Many of you that read this column know that I have been working on a program called Wiz Professional. Well, I did release it at the 'Fest as promised, but not as a commercial product. I decided to go shareware instead. The package is already available on all of the services and many BBS's. If you cannot download it for whatever reason, I will put the same package on a disk that you send me & return that disk to you. If you wish the entire package, see the shareware notice below.

Wiz Professional Sharware Information

Wiz Professional is copyrighted 1988 by William L. Brady.

OS-9 owners who wish to support the development of Wiz Pro and extensions are asked to make donations to: William L. Brady, 1503-I Flanders Lane, Harwood MD 20776. Donations received in excess of \$39.95* will receive the 50+ page laser printed manual, which includes margin notes and many illustrations, plus the unabridged version disks which include WizClipper and WizAuto XPROCS, and other extensions/updates, autolog file examples and source codes.

To become fully registered as a Wiz Pro user, include the following information: First name, Last name, street #, street name, city, state/province, code, country, CIS PIN, Delphi and Genie address.

*Wiz users making a donation in excess of \$19.99 will receive the same package on receipt of a COPY of the WIZ coupon, plus a copy of their sales receipt. See the Wiz manual, do NOT send to FHL.

Those wishing to become Wiz Pro developers will receive additional resources and direct support. A donation of \$25.00 is requested to cover the extra expenses.

Please to not send electronic mail to the author on CIS. Mail is OK on Delphi, (OS9UGED), or GENIE, (B.BRADY), or in the CIS OS-9 forum message base. (Bill Brady 70126, 267).

Should I use this program?

Wiz Pro is a very powerful program. It gets its power from Basic09 and OS-9 Level 2. To do this, it must be fully integrated into the system. There are two reasons why this is important to you. First, you will have to make a new boot. Tandy & Microware do not allow third party s/w writers to provide bootable OS-9 software for the Color Computer. Making a new boot can be tough, but once you master it, you are in control of your own computer. You are the "system manager." The second factor is that you may need to "clean-up" your system. If you have installed patches to system modules, you may find that they interfere with the interrupts that are vital for the RS-232 pak to function. You will need to remove them.

All of the above assumes that you already know about Wiz Pro, but in case you have never heard about it, it is a terminal program that many consider among the best for any computer.

Someone recently referred to me as a third rate programmer. The truth is that I am not a programmer at all. In that regard I am at least 4th or 5th rate (grin). But having been a hardware type back when hardware was expensive and software was cheap, I learned a few important things. Foremost is that I learned how to find "boundaries."

The first of these boundaries is the one between the hardware and the software. To explain why this is so important, I need to use an example, which I will do a little later. First a short course in systems engineering.

This may seem simple, but the first step in deciding on exactly what you want to do with the computer. For example, you want to write a letter. You begin to build a list of requirements. One way is to ask questions like: Does the letter have to be stored in the computer? Is hardcopy required? How big is the letter? Is the letter to contain text only, or text and pictures? Is it necessary to edit the letter?

From this list you can select items and allocate them to resources that are available. Storing the letter might be allocated to a disk drive, printing to a printer etc. As you go along, other requirements build: a

Continued On 48

Modular Programming

Boisy Pitre

Greetings again! I trust you have been practicing some modular design techniques in your work. Things have been rolling and chug-a-lugging here in Brookhaven, Mississippi. As I write this, Hurricane Gilbert, as most of you have heard about, may be planning a visit to our gulf coast in a day or two. Actually, I have been relatively busy squeezing in Disk BASIC programming time between college algebra, trigonometry, and OS-9 Level II. Don't worry fellow Disk BASIC comrades, I haven't been completely converted to OS-9. I'm just dabbling into the fundamentals, but that's another story.

Last issue we went over modular design programming, and some of the events that occur when we neglect to use it. After mastering the BASIC language, we have a tendency to neglect those important aspects of programming we so precariously noted when we first learned BASIC.

Looking back on my first experiences with the CoCo, I can see how my attitude towards preciseness had diminished. Most of the time I had spent planning a simple program was neglected when the biggies came. That's the time I really should have been attentive! After viewing and studying listings from other languages like C and BASIC09, I realized that there was a better way. With eagerness, I purchased a book on structured programming for the IBM. Since GW-BASIC and CoCo BASIC are basically (no pun intended) the same, there were few conflicts with program listings and no conflicts with modular concept.

After studying the book, I realized that complex BASIC programs COULD be done in a truly structured manner. The most dreadful times I faced was when my ideas failed to come about because of complexity in structure. Instead, the confusion of keeping up with variables, GOSUBS, and GOTOS were just too burdensome. Modular programming showed me a better way. And it can help you too. Programming takes as much concentration as it does skill. One can be fluent in BASIC as comparable to an American to English, but enough ?SN ERRORS and ?FC ERRORS can bring frustration to the programmer. Although some make it through, most reach behind the CoCo for the power-switch and push it in, putting the idea out of their head.

This is where modular design steps in. Some of you may have played out the same sce-

nario as the one above. However, programming doesn't have to be a sweat-it-out, mind-boggling experience.

In this issue we will be concentrating on the last two program modules: SUBROUTINES and ERROR TRAPPING. I'll also be giving a few hints and then finally sum up modular programming and announce the "winning idea." For those of you who are just arriving on the scene, I suggest that you order back issue #7 for the beginning article on modular programming.

Ok! Load the MODTUTOR program you typed in last issue. Verify that it is in memory with a LIST or RUN, then type in Listing 1. Don't worry about it conflicting with your program in memory -- the line numbers will neatly fit in place, that is if you haven't RENUMBERED your listing from last month. After you have typed and debugged, SAVE your program under the same name (MODTUTOR.BAS). You now have the complete Modular Design Tutor program available along with a fix for a bug found in the listing last issue. Bear in mind that the purpose for this program was not just to give you something to type. Use it as an outline for your future modular programming projects.

THE ERROR TRAP MODULE -- LINES 40000-49999

This module is debatable. Since the advent of the CoCo 3, error trapping has become a luxury. On the older CoCo 1 and the CoCo 2 however, error trapping is considerably harder to implement and manage. I'll go over both the CoCo and the CoCo 3 techniques of error handling.

CoCo 3 programmers have the luxury of ON ERR GOTO. Somewhere in your MAIN PROGRAM module, set up the line specified by the ON ERR GOTO command in the INITIALIZATION module, which follows this format: EL=ERLIN: ER=ERRNO:GOTO 40000. This way, EL holds the error line and ER holds the error number. Use every two lines, like 40000-40001, 40002-40003, to check the error number and take the needed action, proceeding each with a GOTO EL, or whatever line number you wish. It is also a good idea to comment on each error condition and action with a REMARK.

CoCo 2 programmers don't have it as easy. There are some commercial programs which give CoCo 1 & 2 users an ON ERR GOTO command, but programs of that nature are not owned by all.

VIDEO ENTERTAINMENT

the computer is turned on. (It's up to you to burn the EPROM yourself or find someone to do it for you.)

FKEYS III is a very useful utility, and one which I have enjoyed using a great deal.

I do, however, have a few tips for those of you who have no type of error handling management at all.

One error we will play is the ?NE ERROR, or Non-Existent file. This error occurs mostly when, let's say for instance, one program calls another program or data file on the disk. Use this short program to take care of this error:

```
1000 OPEN"D",#1,"SOMEFILE.DAT"
1001 IF EOF(1)=0 THEN PRINT "FILE DOESN'T
EXIST!":END
1002 CLOSE #1'Continue from here...
```

Your Personal Electronic Photo Album

Put you, your spouse, your kids, your Grandma, your dog, or anything else you like on your very own personal picture disk! You send us your pictures or artwork and we will digitize it and save it on disk for you, in one of the formats below. Originals can be as small as a stamp or as large as a poster. All originals returned.

The cost is \$13.95 for the first disk and \$10.00 for each additional disk, ordered at the same time. Each disk can contain ONE of the following:

- 8 MGE format files (for Colormax3) or
- 10 CM3 format files (for CoCoMax3) or
- 15 PMODE format files (for CoCoMaxII, MCPAINT, etc.)

Pictures should be fairly high contrast or the results can be disappointing. Be SPECIFIC about the roughly square area you want digitized.

Those ordering MGE or CM3 format pictures for their CoCo3 will also receive the incredible viewer program, VUNASTER, absolutely FREE! CoCo1,2 or 3 owners can cut out or copy this ad, send it in with your order and get the \$10 PMODE viewer/manipulator, GrafFind for only \$5! (Reviewed in Apr '88 Rainbow and Jul/Aug '88 CoCo Clipboard)

For a FREE sample disk, please send \$3.00 S/H, along with your model CoCo and the format you wish. Call or write if you have questions or need additional details.

Rainy/Day Software
10825 SE 382nd B32
Boring, OR 97008
(503)663-2423 (9am-3pm PST)

Add \$3 S/H per order.
Payment may be personal check or Money Order only. Sorry, no cash, credit cards or COD's.

DINO PICTURE DATA BASE- 194 Dinosaurs! Names, areas, latin meanings, time periods, pull down menu, world map on screen. Powerful, colorful, educational. COCO3 19.95



Build your own coin-operated Video Game using the COCO! Complete instruction book with plans, schematics, parts sources. All software provided and includes a 4 in 1 video game that you can use. COCO 1 and 2. Disk 19.95

Autoboot your computer- This hardware project will RUN a program as soon as the COCO is turned on, without you having to touch the keyboard! Schematics and instructions only 9.95 COCO1 and 2. Will work on most switch matrix type computers.



FRAZE CRAZE- Wheel a fortune with this popular word game. COCO 1,2 64K or COCO3. 9.95 DISK

THE MEMORY GAME-Flip a card and see if you can find its match. With basic shapes for younger players, all the way up to intricate shapes for those with CONCENTRATING minds. A beautiful family game! COCO3, Joystick, Disk 9.95

Why go to RENO? How you can enjoy the excitement of a Casino style game with BIG CITY KENO! Be a big winner on the COCO3, Disk, 9.95

CHECK, MONEY ORDER OR C.O.D. IN U.S. FUNDS TO RAM ELECTRONICS
814 JOSEPHINE, MONMOUTH, OREGON PHONE NUMBER 503-838-0139



The routine above opened a file called SOMEFILE.DAT. Since this was a random access file, it wouldn't have given an ?NE ERROR if the file didn't exist. Instead, it went ahead and created the file, though empty, and checked to see the EOF flag. If EOF was 0, we automatically knew that the file was empty, therefore never existing in the first place. Of course, you would need to KILL the empty SOMEFILE.DAT file because it was created by OPEN"D".

?SN ERROR is an error that should be encountered with only when developing and testing the program, not at user RUN-time. With experimentation, try finding other ways to circumvent errors. For instance, to prevent a ?DF ERROR (Disk Full), use the D=FREE(0) function. If D=68 or the number of total granules that your disk can hold, then print a DISK FULL message.

THE SUBROUTINE MODULE -- LINES 50000-59999

The SUBROUTINE module contains subroutines for the program. Commonly used routines, such as PRESS A KEY prompts and INPUT commands can be inserted here. I've always believed that it was better to use GOSUBs instead of the same commands over and over again. The SUBROUTINE module benefits the programmer by

conserving memory that would otherwise be used by repetitive commands. It also cuts down considerably on typing.

The SUBROUTINE module should be broken down into separate subroutine fields separated by REM statements. Each REM statement should fully comment the subroutine. I recommend keeping a subroutine library in ASCII format with a line number range of 50000-59999. When you need your library, just MERGE it to the program you are working on. Presto! Instant magic!

SUMMING IT UP

Modular programming is advantageous. It's design and concept is easy to grasp and well organized. Maybe there are some of those who are set in their ways about their own style, but for beginning programmers and those wanting to get into serious Disk BASIC programming, it will save you many wasteful hours. Keep in mind the basic module structure of top-down design. Feel free to experiment with your own preferences. Be innovative! You may be able to program better with another style that the one we have been practicing with. All in all, implement, implement, implement! Use it every chance you get! Learn it to the bone and without a doubt, you will see your ideas come to life!

1040 PRINT:PRINT" THE INITIALIZ
ATION MODULE IS MAKES UP THE FI
RST 100 LINES OF YOUR PROGRAM.

IT IS THE SET-UP PART OF YOUR PR
OGRAM. COMMANDS SUCH AS pclear,
clear, AND dim SHOULD BE USED
HERE.":PRINT

1041 PRINT " YOU MAY ALSO DEFI
NE NUMERIC AND STRING DATA HERE
. USE YOUR 'ONLY-ONCE' COMMANDS
HERE."

1999 GOSUB40001:GOTO 250

2999 GOSUB40001:GOTO 250

3000 '*** CHOICE 3 ***'

3010 CLS

3020 PRINT" PART THREE-ERROR T
RAPPING

3030 PRINT" LINE RANGE:40000-
49999"

3040 PRINT

3050 PRINT" THE ERROR TRAPPING
MODULE IS USED AS AN ERROR DIRE
CTOR. ON THE COCO 3 WITH THE O
N ERR GOTO COMMAND, YOU CAN USE
ERRNO & ERLIN WITH CONDITIONA
L OPERATORSTO TAKE CARE OF ERROR
S EASILY.":PRINT

3060 PRINT" COCO 2 OWNERS SHOUL
D CONSIDER PURCHASING ERROR HAND
LING UTILITIES OR OTHER AL
TERNATIVES."

3070 GOSUB40001:GOTO 250

3999 GOSUB40001:GOTO 250

4000 '*** CHOICE 4 ***'

4010 CLS

4020 PRINT" PART FOUR-SUBROU
TINES

4030 PRINT" LINE RANGE:50000-
59999"

4040 PRINT

4050 PRINT" THE SUBROUTINE MODU
LE HOLDS THE MOST COMMONLY USE
D CODE & STATEMENTS IN YOUR PR
OGRAM. FOR EASY REFERENCE, C
OMMENT EACH SUBROUTINE.":PRI
NT

4060 PRINT " BETTER YET, COMPIL
E YOURSELF A SUBROUTINE LIBRARY
WITH YOUR MOST COMMONLY USED S
UBROUTINES. SAVE IT IN ASCII, AN
D MERGE IT TO THE PROGRAM YOU A
RE WORKING ON."

4070 GOSUB40001:GOTO 250

4999 GOSUB40001:GOTO 250

5999 GOSUB40001:GOTO 250

40000 '***** SUBROUTINES *****'

40001 'PRESS A KEY PROMPT

40002 PRINT@490,"PRESS A KEY: ";:
EXEC34442:RETURN

1040 PRINT:PRINT" THE INITIALIZ
ATION MODULE IS MAKES UP THE FI
RST 100 LINES OF YOUR PROGRAM.

IT IS THE SET-UP PART OF YOUR PR
OGRAM. COMMANDS SUCH AS pclear,
clear, AND dim SHOULD BE USED
HERE.":PRINT

1041 PRINT " YOU MAY ALSO DEFI
NE NUMERIC AND STRING DATA HERE
. USE YOUR 'ONLY-ONCE' COMMANDS
HERE."

1999 GOSUB40001:GOTO 250

2999 GOSUB40001:GOTO 250

3000 '*** CHOICE 3 ***'

3010 CLS

3020 PRINT" PART THREE-ERROR T
RAPPING

3030 PRINT" LINE RANGE:40000-
49999"

3040 PRINT

3050 PRINT" THE ERROR TRAPPING
MODULE IS USED AS AN ERROR DIRE
CTOR. ON THE COCO 3 WITH THE O
N ERR GOTO COMMAND, YOU CAN USE
ERRNO & ERLIN WITH CONDITIONA
L OPERATORSTO TAKE CARE OF ERROR
S EASILY.":PRINT

3060 PRINT" COCO 2 OWNERS SHOUL
D CONSIDER PURCHASING ERROR HAND
LING UTILITIES OR OTHER AL
TERNATIVES."

VIP Calc

"MORE USEABLE FEATURES"
FEBRUARY 1985 "RAINBOW"


Now every CoCo owner has access to a calculating and planning tool better than VisiCalc™, containing all its features and commands and then some. VIP Calc displays 32, 51, 64 or 85 characters by 21 or 24 lines right on the screen. VIP Calc allows up to a 33K worksheet with up to 512 columns by 1024 rows! In addition, VIP calc has multiple windows which allow you to compare and contrast results of changes. Other features include 16 DIGIT PRECISION • trig. functions • averaging • algebraic functions • column and row ascending or descending SORTS • locate formulas or titles in cells • block move and replicate • global or local column width • limitless programmable functions • works with any printer, Embed printer control codes for customized printing. Combine spreadsheet tables with VIP Writer documents to create ledgers, projections, statistical and financial budgets and reports. Requires 64K. **DISK \$59.95**

VIP Speller 1.1

INCLUDES 50,000 WORD DICTIONARY
VIP Speller works with ANY ASCII file created by most popular word processors - even Telewriter 64. It automatically checks text files for words to be corrected, marked for special attention or even added to the 50,000 word Dictionary. You can even view the word in context. Words can be added to or deleted from the dictionary or you can create your own dictionary! New features of version 1.1 are **FASTER** and more **RELIABLE** disk access and printing at 9600 BAUD. **DISK \$34.95**

VIP Speller owners: upgrade to VIP Speller 1.1 for \$5 + \$3 S/H. Send only ORIGINAL disk and \$8 Total.





EZ FRIENDLY For Your CoCo 1, 2, & 3

EZWriter

"EZWRITER" does professional-quality letters quickly and easily. "EZWRITER" will automatically set the margins & characters per line, indent paragraphs, put your heading/greeting/closing in the right places, and even center the letter on the page! Just type your message and watch your DWP or DMP print a perfect letter!!! Handles 1 - 4 page letters, mailing lists, and labels. It will make it possible for you to send any number of copies with a salutation for each recipient!! Perfect for personal correspondence, for letters to club members, or for direct mail advertising. "EZWRITER" is by far our #1 Best Seller, and it's no wonder; it's the absolutely simplest letter writing system available for the CoCo! Still only \$19.95 plus \$1.50 S/H for either disk or cassette. GREAT on the CoCo 3 but will run on any CoCo with 32k ECB.

E.Z. FRIENDLY SOFTWARE
 118 CORLIES AVE. • POUGHKEEPSIE, NY 12601 • (914) 485-8150
 (Add \$1.50 s/h to all orders. NY residents add state sales tax.)

capability of opening, reading, modifying disk files is needed, so we allocate that to the software. When you store files on a disk, they must be named... you allocate that to the operator (perhaps). The files should also be organized in some way so they are easy to find.. you allocate that to...software? the operator? the operating system?? Beginning to get the idea?

Now for the importance of this process of allocation. Back in the 70's computer hardware was not as smart as it is today. The math capabilities were often limited to add, shift, AND, OR, and complement. You subtracted by doing a complement, then an add. Multiply was done by shifting. About this time I worked a project where there was a real need for the capability for high precision (floating point) math. The capability did not exist. Since the person in charge was a programmer, this requirement was allocated to s/w. A number of programmers started work.

A great deal of time and money were used in this effort. A deadline loomed. The floating point software package still did not work, it was very slow, it used up a great deal of hardware resources. Perhaps another computer would be required. Enter the hardware types.

We found a black box called a floating point processor. It cost \$5000, less than 1/50th of what had already been spent on the s/w effort. It was lightning fast, and used almost none of the main processors time since

it ran in parallel. It had one other advantage: it worked.

So you can see that finding that hardware / software boundary was very important. If it had been done sooner it would have saved many dollars, later would have meant the failure of the project.

These boundaries exist, and must be identified at many levels. There are boundaries between h/w and s/w, between h/w and operations, between s/w and operations, and between s/w and s/w. Software boundaries include those between the operating system and the application, between an application and a device driver and between the operating system and a device driver.

Boundaries shift. What causes these shifts? Time, money, the operator, and the building of the resource pool in both s/w and h/w. In what directions do we want the shifts to go? Less time, both to develop, and to use, and less money of course. But the never varying final goal is to shift the boundary AWAY FROM OPERATIONS!

How does all of this apply to Basic09? Remember that my goal in this column is not just to tell you how to do something, any programmer can do that, but to tell you the why and when. Why use Syscall? When? Why use Basic09? When? Let us begin to take a look at the boundary between a piece of hardware, a modem, and a piece of software, a terminal program.

FILE TRANSFER UTILITIES

You asked for it at the Chicago RainbowFest -

FILE TRANSFER UTILITIES NOW HANDLE RSDOS DISKS!

Need to transfer text files to and from PC (MSDOS), RSDOS and FLEX disks into your CoCo (OS-9) system? Have text files on a PC (MSDOS) system at work and want to work on them at home on your CoCo?

With GSC File Transfer Utilities you just place the PC (MSDOS), RSDOS or FLEX disk into your CoCo disk drive - enter a simple command and the file is copied into a CoCo OS-9 file. File transfer back to PC (MSDOS), RSDOS and FLEX disks is just as simple.

PCDIR directory of PC disk PCDUMP display PC disk sector PCREAD read PC file PCWRITE write file to PC disk	RSDIR directory of RSDOS disk RSDDUMP display RSDOS disk sector RSREAD read file from RSDOS disk RSWRITE write file to RSDOS disk
PCRENAME rename PC file PCDELETE delete PC file PCFORMAT format PC disk	FLEXDIR directory of FLEX disk FLEXDUMP display FLEX disk sector FLEXREAD read FLEX file FLEXWRITE write file to FLEX disk


Extensive Options Single, double sided disks. 40 or 80 track floppy drives. 8 or 9 sectors. First level sub-directories - PC (MSDOS). FLEX transfers binary files also.

Requires OS-9 (Level 2 for MultiVue), 2 drives (one can be hard), MultiVue for MultiVue version, SDISK (SDISK3 for MultiVue) - see D.P. Johnson ad for SDISK

GSC File Transfer Utilities for CoCo - MultiVue version \$54.95

GSC File Transfer Utilities for CoCo - Standard version \$44.95

All diskettes are CoCo OS-9 format. Orders must be prepaid or COD, VISA/MC accepted, add \$1.50 S&H, additional charge for COD.

GRANITE COMPUTER SYSTEMS 
 Route 2 Box 445 Hillsboro, N.H. 03244
 (603) 464-3850

OS-9 is a trademark of Microware Systems Corporation and Motorola Inc.
 MS-DOS is a trademark of Microsoft Corp. FLEX is a trademark of TSC, Inc.

Data comes out of a modem serially, that is one bit at a time. The computer stores data in parallel, 8 bits per word. If we are to move data out of the modem and into the computer, a serial to parallel conversion must take place. In the Color Computer we have a hardware device called a PIA. Without going into great detail I will just say that the PIA can be made to read one bit at a time. With software we can convert 8 bits to one word, but it is not that simple. You must know which 8 bits to put into any particular word - it's called synchronization. All of this can be done. Many fine terminal programs exist which do this technique called "bit banging." The problem is that this process takes every bit of the hardware's resources, just to work at relatively low baud rates. The bottom line is that you cannot do anything else while the conversion is going on and even then it is slow at it's max speed.

Remember the floating point hardware? Well there is "serial-to-parallel" hardware also. One of these devices is called an Asynchronous Communications Interface Adapter or ACIA for short. Just like the floating point h/w, the ACIA is much faster, and takes no resources and does both the serial to parallel conversion and the synchronization. Unfortunately, Tandy did not build an ACIA into the Color Computer.

So, if we are writing a terminal program for the CoCo, we must decide whether to allocate the s/p conversion and synchronization capabilities to h/w (an RS-232 Pak) or to the s/w (a "bit banger".) Not all users own an RS-232 pak.

Under OS-9 our decision is a little easier. OS-9 is multi-tasking and on the CoCo we have windows. These two capabilities of the operating system allow us to clear-key to another window and run some other program at the same time as our terminal program. Those of you who use this capability know that this is no trivial nicety, but is what makes the CoCo such a powerful computer today. Why throw this away? Why not xmodem a file in one window and de-archive it on the fly in another?

To use the bit-banger port would negate all of this. So we allocate the capability to h/w and require an RS-232 pak. This still doesn't solve the problem however, and we can still blow our multi-tasking capability even with an RS-232 pak. How? By placing yet other boundary in the wrong place! Next time we will see just how fine this line must be cut, and look at some code.

Before leaving you, however, I promised to show you how to capture a mouse "key down" event within a Basic09 procedure. With Wiz Pro, any time you are on line you can pause both Wiz and the host by simply tapping the mouse key. This allows you to stop something from scrolling off of your screen before you've had a chance to read it. For this to happen, and to really be useful, everything must come to a halt very quickly. I choose to use the SS.MsSig system call.

The SS.MsSig call tells OS-9 to send you a signal when the mouse button is pressed. The

call also allows you to specify a signal # for the call to return. See page 8-141 in the Level 2 manual.

First lets set the signal:

```
(* *****
s.a=0\ s.b=$8A\ s.x=140
RUN syscall(ststat,s)
(* *****
```

Now something to "catch" the signal:

```
(* *****
ON ERROR GOTO 100
(* *****
```

Now something to process the signal:

```
(* *****
100 en=ERR
IF en=141 then
(* send x-off, go to sleep etc
(* *****
```

When we are we don't want to leave the signal active so we disable:

```
(* *****
s.a=0\ s.b=$1B \ s.x=140
RUN syscall(ststat,s)
(* *****
```

See you the next time! -Bill Brady

VIP TERMINAL

VIP Terminal

**RATED BEST IN JANUARY
1984 "RAINBOW"**

For your important communications needs you've got to go beyond software that only lets you chat. You need a smart terminal so that you can send and receive programs and messages and print them! The VIP Terminal features 32, 51, 64 or 85 characters by 21 or 24 lines on the screen and has a 43K byte buffer to store information.

DISK \$29.95

VIP DISK-ZAP

VIP Disk-ZAP 1.1

**RAVED ABOUT IN THE
APRIL 1983 "RAINBOW"**

Now you can retrieve lost data on any disk. VIP Disk-Zap is the ultimate repair utility for repair of most disk errors. VIP Disk-Zap verifies diskettes, reads and writes any sector and lets you retrieve all types of bashed text files, BASIC and ML programs. VIP Disk-Zap includes an informative 50 page tutorial manual. New features of version 1.1 are FASTER and more RELIABLE disk access and printing at up to 9600 BAUD.

DISK \$24.95

VIP DISK-ZAP

VIP Disk-Zap owners: upgrade to VIP Disk-Zap 1.1 for \$5 + \$3 S/H. Send only ORIGINAL disk and \$8 Total.

CoCo and The Model 100

Rush Caley

I was dumbfounded when Ted asked me to do an article about this subject. I've been relatively well known in the CoCo market for a number of years - at least in the business applications portion of it. But one thing not many people know about me is my passion for the Model 100 laptop computer. It is not as though I have to keep it sequestered away somewhere so no one will notice it; but I don't talk about it too much. There's a certain "cloak of loyalty" that has to be worn and recognized when facing CoCo fanatics. It's funny, I use my Model 100 almost as much as I do the CoCo. And for all these years, I still feel like a philandering husband cheating on his faithful wife.

Anyone who has used a laptop for any length of time will realize that how you support the laptop is wholly defined by your specific "workstyle" and the particular tasks you need to accomplish. The purpose of this article is to mention some of the more useful things that one can do by interfacing the Model 100 with our beloved CoCo.

When it was originally released, the M-100 arrived "out of the box" with 24K RAM and sported an 80C85 8 bit microprocessor that ran at 2.4 MHz. The LCD screen was 40 X 8 lines. It also came loaded with ROM software or "built-in" applications programs. It had BASIC (one better than the CoCo's); TELCOM, telecommunications capabilities at home or from any phone booth; a TEXT processor, albeit limited; and two other applications whose primary uses were maintaining addresses and scheduling appointments and things-to-do. It also had parallel printer port, RS-232, a Bar Code reader, a terrific keyboard, and much more. But the best thing about this highly touted Tandy toy, was that it fit into any standard briefcase and you could compute anyplace you happen to be. The price was: \$999.95.

Originally, the M-100 was designed to store its data on tape cassette. You long time CoCo users know what that is like. So the very first obstacle to overcome was storage capacity and more speed. Early on, Tandy offered its Disk Video Interface, but it cost \$800! Also, it was clumsy to hook up and unhook the darn thing. Furthermore most people bought the M-100 for its portability, and the DV/I seemed to run contrary to intent. Then a company in Dallas came up with a unique concept. They called the program DISK+. It

required a null modem cable and had 2 functions. First, it would allow you to hook up to your CoCo via the serial port, and store programs and files on your CoCo's drives. Also, it could unload or load them at 9600 baud! Additionally, it had a file transfer program so that you could convert the files back and forth from Model 100 format or CoCo format. For example, if I were on the road, and picked up a nifty program off the CoCo SIG on Compuserve, I could download it to my M-100 and then upload it to the CoCo when I returned home. So this was my first use for interfacing the CoCo with my portable: storage space, and speed of file saving and loading.

This interface ability became a real boon in the near future. I had moved from Seattle to my current residence in Port Orchard and had to commute back and forth on the ferry boat every day. Fortunately, I was able to use the commute time productively. I could type away on my portable and then transfer files to the CoCo when I arrived at work. I kept my time management and scheduling program on the model 100 so I always had the information at hand. The files could be stored on the CoCo at work and on my M-100's "Chipmunk" disk drive at home. I don't think I could count the number of documents polished up on Telewriter 64tm that had originated on my M-100. The Spreadsheet I used is called LUCIDtm. It is a very sophisticated program with capabilities I did not have on the CoCo. I could store spreadsheet files on the CoCo disk. Also, I could spool the formatted output to an ASCII RAM file and read it into Telewriter at work. These and other applications like them, made my commuting days much more tolerable.

THE PRESENT

The "CoCo/Pico" romance was now in full bloom and has continued ever since. In my present job at Boeing Commercial Airplanes, I am not allowed to bring my laptop to work. Now I have a separate workstation in my office at home and have it hooked up to a Disk Video Interface. (I got it cheap!) I've increased the RAM to 96K and have 2 drives. Following are some of the specific uses I now have for the Model 100 and CoCo working together and some of the advantages.

I.) From a telecommunications point of view,

using the laptop for all these tasks has definite advantages. Whether I am at home or away, I can always check my electronic mail boxes, attend on-line conferences, download programs and files for either machine, and avoid the limitation of needing to be home at the CoCo.

II.) One of the niftier little file management programs I use for a variety of uses on the CoCo is JULIE THE MOUSE. This freeform file manager has the ability to output JULIE file formats to an ASCII file that can be sent via modem. Anyone having the program can download the file. Another way the laptop assists in JULIE applications is that I do not have to be home at the CoCo to initiate a new file. I can type in my records on the M-100 in JULIE format. Then when I return home, I can upload the file to the CoCo.

III.) One of the most unfortunate missing applications for the CoCo is that of an outliner. If you have never heard of one or used one, you don't know what you're missing! I use a model 100 program called IDEA! by Traveling Software INC. It is a program I cannot live without. In any case, I spool the IDEA! file formatted to a RAM file and then upload it to the CoCo and BUSINESS WRITER™ for cleanup and final formatting.

IV.) I still use LUCID spreadsheet and move the files up to the CoCo when necessary.

THE FUTURE

Hopefully, the capabilities and uses for the CoCo / Model 100 interface will increase. I have a programmer friend in Seattle who has an excellent client mailing list program he wrote for his CoCo. He also has a Model 100 so he wrote a model 100 version of his mailing program and is working now on a method to allow the files to be read by either machine. He has also developed a special null modem cable that "tricks" the CoCo into thinking that the Model 100 is a printer. This allows him to send data files to the Model 100 from CoCo BASIC without the use of a terminal program.

And finally, there is another undeveloped use for this setup that already exists between the Model 100 and the PC compatibles. This is the ability to have a "host program" that allows the laptop user to call up his desktop from a remote location and actually run PC programs. This would be an excellent project for a talented CoCo programmer!! If I could "run" CoCo programs from a remote location using my laptop, it would be a user's dream come true!

If you are a CoCo user who has a need for some of these applications and abilities, a Model 102 laptop might not be a bad investment. You might want to do it just for the fun! If you're a wife who considers yourself a "CoCo Widow", consider yourself lucky! Think how it could be if the darn CoCo

followed you no matter where you went!. Ask my wife, she used to go crazy because I'd pack up my Model 100, a 360K disk drive, and a thermal printer and take it everywhere we go! All of this stuff runs on batteries and fits in a briefcase. But you work it out - I'm not a marriage counselor.

POKE 113,0: EXEC40999

Finally

An answer for America's dying teens!
For your free copy of this dynamic and powerful book, call us today at
1-800-VICTORY.



Also;
See and hear young people who have already found the answer at Freedom Village, USA in Lakemont, N.Y.

Check your local Christian radio and television stations for the Victory Today program.
For more information write Freedom Village, USA Lakemont, NY 14857

VIP Database 1.1

"ONE OF THE BEST" JUL '84 "RAINBOW"

VIP Database has all the features of VIP Database III described elsewhere in this magazine except the screen widths are 51, 64 & 85. Screen colors are black, green and white, double clock speed and Spooler are not supported. Even so, VIP Database is the most complete database for the CoCo 1 & 2! Version 1.1 has FASTER and more RELIABLE disk access and single spaced reports. **DISK \$49.95**

VIP Database owners: upgrade to VIP Database 1.1 for \$5 + \$3 S/H. Send only original disk and \$8 total.

VIP Writer 1.1

RATED "BEST" IN SEPT '88 "RAINBOW"

VIP Writer has all the features of VIP Writer III described elsewhere in this magazine except the screen widths are 32, 51, 64 & 85. Screen colors are black, green & white, double clock speed is not supported, Spooler and menus are unavailable because of memory limitations. Even so, VIP Writer is the BEST word processor for the CoCo 1 & 2! Version 1.1 includes the configuration/ printer installation program and RGB Hard Disk support. Available thru Radio Shack Express Order #90-141.

Includes VIP Speller 1.1 DISK \$69.95

VIP Writer owners: upgrade to VIP Writer 1.1 for \$15 + \$3 S/H. Send only ORIGINAL disk and \$18 total.



Reader Mail



Dear Editor:

Since SD Enterprises has taken over the VIP line of Color Computer software we have uncovered a problem with the way some of the VIP programs work with slower disk drives. It seems that the VIP Software does not allow sufficient time for some disk drives to come up to speed before reading or writing data.

This problem has been totally solved by restoring the delay built into RSDOS within the VIP Software. The VIP Writer III version 2.0 and VIP Database III do not have this problem.

New versions of the original VIP Writer, Speller, Database and Disk-Zap for the CoCo 1 and 2 with this enhancement are available for a small fee. These new versions also add additional features. Please see our ads for more details.

Paul Anderson
S D Enterprises

Thanks Paul for this updated information. Our readers are urged to watch for an updated review of VIP Writer III Version 2.0

Dear Mr. Paul:

My compliments to you on the direction you are taking CoCo Clipboard! You're doing a splendid job.

I was especially interested in Boisy Pitre's article "Modular Programming". Excellent! It should be emphasized that regardless of what language you're using for programming (and this includes data-bases and spreadsheets), modular programming makes life much easier on all concerned and reduces logical programming errors.

One suggestion I'd like to make regarding line numbers and interpreted Basics such as RS Basic. A significant improvement in speed can be obtained if the program organization suggested is altered. Place initialization statements at the end of the program and frequently used subroutines at the beginning of the program in the order of frequency of use. Put the main module somewhere in the middle. Subroutines used infrequently may be placed between the main module and the initialization statements. The first line of the program would contain only pclear and pmode statements, as required, immediately followed on the same line by a goto (Initialization Module) at the tail of the program. If the modular concept is carried to

its limit, the main module will consist only of "gosub" statements.

This speed advantage (or loss) is due to the nature of most interpreted basics. Every time a line number is called, the interpreter goes back to the first line number, jumps to that line number, calculates the "distance" to the next line number, etc. until it reaches the desired line number. Putting initialization routines and other seldom used routines at the beginning of the program only slows everything down.

A tip regarding remarks. As urged by Mr. Pitre, use them liberally. Frankly, if I didn't have remarks, I wouldn't know what I did five minutes ago. But remarks slow down a program (have to step through those line). A better practice is to save and print a copy of the program after debugging. The delete all the remarks and renumber the line numbers stepping the line numbers by 1. This reduces the size of the program drastically and further speeds execution. If you want to make changes, they are usually more easily made to the original program followed by the delete and renumber process.

Just my two cents - hope it helps. Again, keep up the excellent work.

Ed Gresick
Delmar Co.
Middleton, DE

As many of you may remember it was just one year ago that we ran an article called "The Dream Machine" which featured Ed's CoCo Computer system and how it was used to run his Radio Shack Franchise store. That issue is still available in reprint form.

In addition Ed has become one of our biggest supporters and joined with Darlene and myself - and Jim DeStafeno at the Philadelphia Area Computer Society CoCo meeting in September. Ed has many years of programming experience and if you're in the Middleton Delaware area stop by and say hi to Ed and watch his CoCo business system go!

To: Mike Dooley:

I have been reading your Ham radio column, and I hope we will get some new interest in Ham radio from the computer people reading the magazine.

Anyhow perhaps you can mention a few things in your column:

1) The ARRL has a program exchange service, for an SASE they will send you a

listing. Readers should write to the ARRL, Dept. PX, 225 Main St., Newington, CT. 06111. Include the SASE and ask for a list of programs.

2) For Packet Radio try "Auto-Term"

3) Check out STV pictures, no interface receive only using Graphicom.

There is also a book from Radio Shack entitled "Electronic Programs for Circuit Designs", I don't have the catalog number but it only costs \$4.95 and is a great bargain.

I keyed in the ARRL Dept. PX programs, found out 2 of them have errors in them, as well as the programs from the Radio Shack book. I got the CoCo for Ham radio purposes and enjoy using it in the shack with the radio and as a stand alone computer.

73's
Dave

(From Mike Dooley) Thanks for the letter and all of the good information! I'll be checking out the ARRL's program exchange along with the other programs you mentioned. I got my CoCo for the same reason as you. I kept finding all of these neat interfaceless programs for the CoCo and had to try them! I'll be covering as many as possible in the upcoming months (bi-months?) along with listings to be typed in. Hang in there, Dave ... Ham Radio is growing!

Dear Publisher:

Enclosed please find a check for the back issues mentioned in your flyer.

If the magazine is as good as you say it is you can expect to see a subscription or two from this part of the country. I took your letter to our local CoCo club meeting to see what interest there would be. Everyone was enthusiastic but skeptical. Sure hope this is the POT OF GOLD at the end of the RAINBOW. A lot of us are getting tired of the CoCo being thought of as a kid's toy!

Thank you in advance for what we hope will be a shot in the arm for the CoCo Community in the Atlanta area.

Yours
Alan R. Dages
Stone Mountain, GA

Dear Alan: Thanks for the support from your part of the country. We're working hard to promote the CoCo as a serious computer and our growing subscription base of over 800 PAID is the proof. We can always use more so please spread the word!

Dear Ted and Darlene:

Enclosed is a money order for another year of your much needed magazine.

I recently bought Multi-View, and am STILL having trouble trying to revise the "bootlist" file, so that I may take advantage of my 720K drive. Bob van der Poel stated in his article "New Boots" (July/August '88) that it took him about an hour trying to make his

custom "bootlist". That would be Wonderful if that's all it took. Being "not-so-affluent" with OS-9, I've been working on this problem for a good solid 96 hours total, and am still messing with it. He's right - much too much HAS been written on this subject. So much, in fact, that there is no ONE place a person can go to for in-depth instruction.

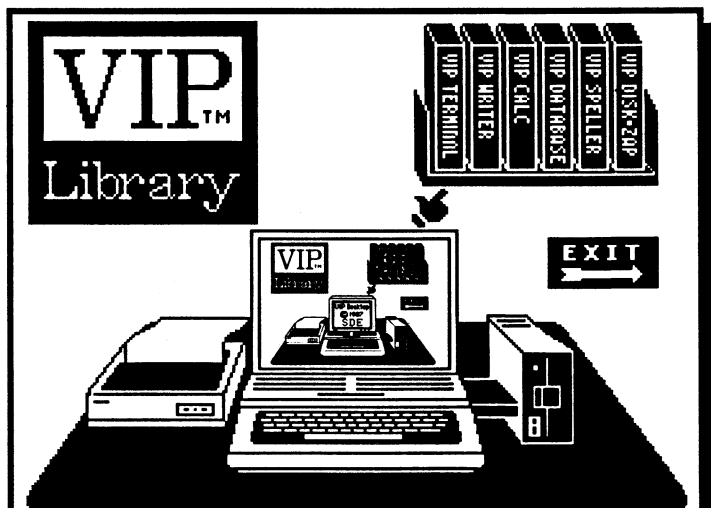
Another thing that nobody seems to be willing to fully explain, is if I can use the 720K drive (now drive 2) as the default drive.

Anyway, keep those issues coming, and I'll continue pounding away, and maybe if I hit all the right keys in the correct order one day, all this will seem foolish to me!

Thank You
Joe Arntz
Poughkeepsie, NY

Well Joe, here's your letter and I hope our readers will respond quickly to your needs. A source you might want to check out is the OS-9 SIG on CompuServe. I've received a lot of help and information on CIS and perhaps we will get one of the OS9 experts to be our on line guests during our monthly conferences.

Well I wish we had more space for letters this month. As usual we love to hear from you by phone, letter or on CIS.



The VIP Integrated Library 1.1 combines all six popular VIP programs - Writer 1.1, Speller 1.1, Calc, Database 1.1, Terminal and Disk-Zap 1.1 - into one program on one disk. The program is called VIP Desktop. From the desktop you have instant access to word processing with a spelling checker always in attendance, data management with mailmerge, spreadsheet financial analysis, telecommunications and disk maintenance. Just move the hand to the volume on the bookshelf and the application is there. 64K req'd. \$149.95 + \$4 S/H US, \$5 Canada, \$10 Foreign. VIP Library is available thru Radio Shack Express Order #90-213.

VIP Integrated Library owners: upgrade to the VIP Integrated Library 1.1 for \$25 + \$3 S/H. Send only ORIGINAL disk and \$28 Total.

SD ENTERPRISES

(503) 663-2865 P. O. BOX 1233. Gresham, OR. 97030
Non VIP Library orders add \$3 for shipping in USA, Canada \$4,
Foreign \$6. COD orders add an additional \$2.25. Personal checks
allow 3 weeks for delivery. All other orders shipped the same day.

VIP Database III



"...a remarkable job in creating a nicely updated version of a tried and true CoCo database. ...this program just roars! Database III...a high powered productivity worker in your home or office."

-CoCo Clipboard Sept. 1988

- ✓ Supports 40, 64 and 80 column hardware screen.
- ✓ Separate text and highlight colors for easy data entry.
- ✓ Built-in floating point MATH package.
- ✓ IN-MEMORY sort of all records for maximum speed.

- ✓ Built-in mailmerge prints form letters & mailing labels!
- ✓ Print spooler lets you print while you enter more data.
- ✓ Stores 558 records of 256 bytes on a 1 disk system.
- ✓ Multi-criteria search with up to 16 relational keys.

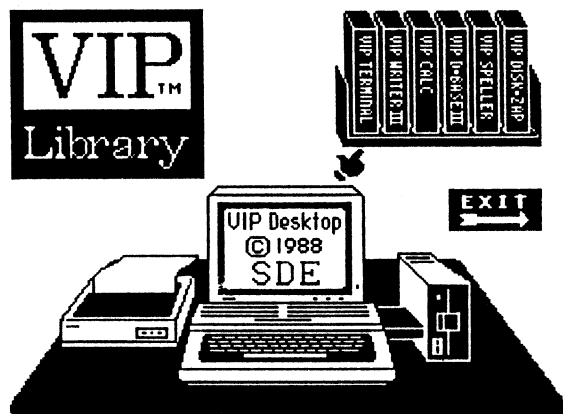
VIP Database III features selectable screen displays of 40, 64 or 80 characters by 24 lines with choice of 64 foreground, background, hilite and cursor colors for EASY DATA ENTRY. It uses the CoCo 3's hardware screen and double clock speed to be the FASTEST database available! VIP Database III will handle as many records as will fit on your disks and is structured in a simple and easy to understand menu system with full prompting for easy operation. Your data is stored in records of your own design. All files are fully indexed for speed and efficiency. IN-MEMORY SORT of records is LIGHTNING FAST and provides for easy listing of names, figures, addresses, etc., in ascending

or descending alphabetical or numeric order. Records can be searched for specific entries using multiple search criteria. The built-in mailmerge lets you sort and print mailing lists, print form letters, address envelopes - the list is endless. The built-in MATH PACKAGE even performs arithmetic operations and updates other fields. VIP Database III also has a print spooler and report generator with unlimited print format capabilities including embeddable control codes for use with ALL printers.

DISK \$69.95

VIP Database owners: Upgrade to the VIP Database III for \$39.95 + \$3 S/H. Send ORIGINAL disk and \$42.95 total.

VIP Integrated Library



Merry Christmas

/Writer Database Enhanced

The VIP Library /WDE combines all six popular VIP application programs - VIP Database III, VIP Writer III, VIP Speller 1.1, VIP Calc, VIP Terminal and VIP Disk-ZAP 1.1 - into one program on one disk! The program is called VIP Desktop. From the desktop you have instant access to word processing with a spelling checker always in attendance, data management with mail merge, spreadsheet financial analysis, tele-communications and disk maintenance. *

DISK \$169.95

For VIP Library shipping please add \$4 USA, \$5 Canada, \$10 Foreign.

VIP Library owners: Upgrade to the VIP Library /WDE for \$89.90* + \$3 S/H. Send ORIGINAL disk and \$92.90 total.

* Future VIP Library upgrades available at reduced cost.

All products run under RSDOS and are not copy protected.

SD ENTERPRISES

© (503) 663-2865 ☐ POB 1233 Gresham, OR 97030

Non VIP Library orders add \$3 for shipping and handling in USA, Canada \$4, Foreign \$6. COD orders add an additional \$2.25. Checks allow 3 weeks for delivery. All other orders are shipped the same day.

VIP Writer III 2.0

495K Total Text Space • EASY 4 Color MENUS



COMPARISON CHART			
	VIP Writer III	Telewriter 128	Word Power 3.2
Screen Display	32/40/64/80	40/80	80
Spelling Checker	VIP Speller	NONE	FREE WARE
Dictionary Size	50,000 Words	NONE	20,000 Words
RGB HD Support	100%	N/A	NO
Print Spooler	YES	NONE	YES
Total Space 128K	106,000	48,000	72,000
Total Space 512K	495,000	48,000	450,000

N/A - Information is Not Available. Numbers are rounded to the nearest thousand.

MORE SCREEN DISPLAY OPTIONS

VIP Writer III offers more screen width options -all with 24 lines and actual lower case letters using the CoCo 3's hardware display. It runs at double clock speed and has 4-color menus making VIP Writer III FAST and EASY to use! You can choose foreground, background, hilite and cursor colors from up to 64 hues. Color can be turned ON or OFF for the best possible display using a monochrome monitor or TV set. VIP Writer III has a built in on-line context sensitive help facility to display command usage in easy to read colored windows.

CUSTOMIZER & PRINTER INSTALLER

VIP Writer III comes with a configuration / printer installation program which lets you customize VIP Writer III to suit your own liking. You can set screen width and colors as well as margins and more. You can also install your own printer and set interface type (serial, parallel or J&M), baud rate, line feeds, etc. Once done, you never have to enter these parameters again! VIP Writer III will load n' go with your custom configuration every time!

ASCII TEXT FILE STORAGE

VIP Writer III creates ASCII text files which are compatible with all other VIP Programs as well as other programs which use ASCII files. You can use VIP Writer III to even type BASIC programs! There is a 48K text buffer (438K in a 512K CoCo 3) and disk file linking allowing virtually unlimited text space. VIP Writer III works with up to 4 disk drives and lets you display directories and free space as well as rename or kill disk files. In addition VIP Writer III supports double sided drives using RGB-DOS in ROM and is 100% compatible with RGB Computer Systems HARD disk.

"...High powered software for the CoCo3...Fast, easy to use...a true bargain for the serious CoCo user"
-CoCo Clipboard MARCH 1988

POWERFUL EDITING FEATURES

VIP Writer III has a full featured screen editor which can be used to edit text with lines up to 240 characters long with or without automatic word wrap around. You can select type-over mode or insert mode. There is even an OOPS command to recall a cleared text buffer. Other editing features include: Type-ahead • typamatic key repeat and key beep for flawless text entry • end of line bell • full four way cursor control with scrolling • top of textfile • bottom of textfile • page up • page down • top of screen • bottom of screen • beginning of line • end of line • left one word • right one word • DELETE character, to beginning or end of line, word to the left or right, or entire line • INSERT character or line • LOCATE and/or CHANGE or DELETE single or multiple occurrence using wildcards • BLOCK copy, move or delete with up to TEN simultaneous block manipulations • TAB key and programmable tab stops • word count • line restore • three PROGRAMMABLE FUNCTIONS to perform tasks such as auto column creation and multiple copy printing.

AUTOMATIC TEXT FORMATTING

VIP Writer III automatically formats your text for you or allows you to format your text in any way you wish. You can change the top, bottom, left or right margin and page length. You can set your text flush left, center or flush right. You can turn right hand justification on or off. You can have headers, footers, page numbers and TWO auxiliary lines which can appear on odd, even or all pages. You can also select the line on which they appear! You can even change the line spacing! Parameters can be altered ANYWHERE within your text!

"In the beginning there was VIP Writer and users saw that it was good, But it's not the best anymore. There's a new word processor to claim the crown...VIP Writer III -Setting the Standard"
-RAINBOW Sept. 1988

PREVIEW PRINT FORMAT WINDOW

VIP Writer III features an exclusive format window which allows you to preview your document BEFORE PRINTING IT! You are able to move up, down, left and right to see centered and justified text, margins, page breaks, broken paragraphs, orphan lines etc.

PRINTING VERSATILITY

VIP Writer III prints TWICE as fast as any other CoCo word processor! It supports most serial or parallel printers using J&M JFD-CP or Rainbow interface and gives you the ability to select baud rates from 110 to 19,200. You can imbed printer control codes anywhere in your text file EVEN WITHIN JUSTIFIED TEXT! VIP Writer III also has TWENTY programmable printer macros which allow you to easily control all of your printers capabilities such as bold, underline, italics and superscript using simple key strokes. Other features include: multiple copy printing • single sheet pause • add line feeds • print pause to change print wheel.

BUILT IN PRINT SPOOLING

Save up to \$150 on a print spooler because VIP Writer III has a built in print spooler with a 57,000 character buffer which allows you to print one document WHILE you are editing or proof reading another. You don't have to wait until your printer is done before starting another job!

50,000 WORD SPELLING CHECKER

VIP Writer III includes VIP Speller AT NO ADDITIONAL COST! VIP Speller checks text for misspelled words and has a 50,000 word dictionary that can be added to or edited.

QUALITY DOCUMENTATION

VIP Writer III comes with a well written 125 page manual which is Laser printed, not dot-matrix like the competition. It includes a tutorial, glossary of terms and examples for the beginner as well as a complete index! VIP Writer III is truly the BEST you can buy.

Includes VIP Speller 1.1. **DISK \$79.95**

VIP Writer owners: Upgrade to the VIP Writer III 2.0 for \$49.95 + \$3 S/H. Send ORIGINAL disk and \$52.95 total.

SD ENTERPRISES

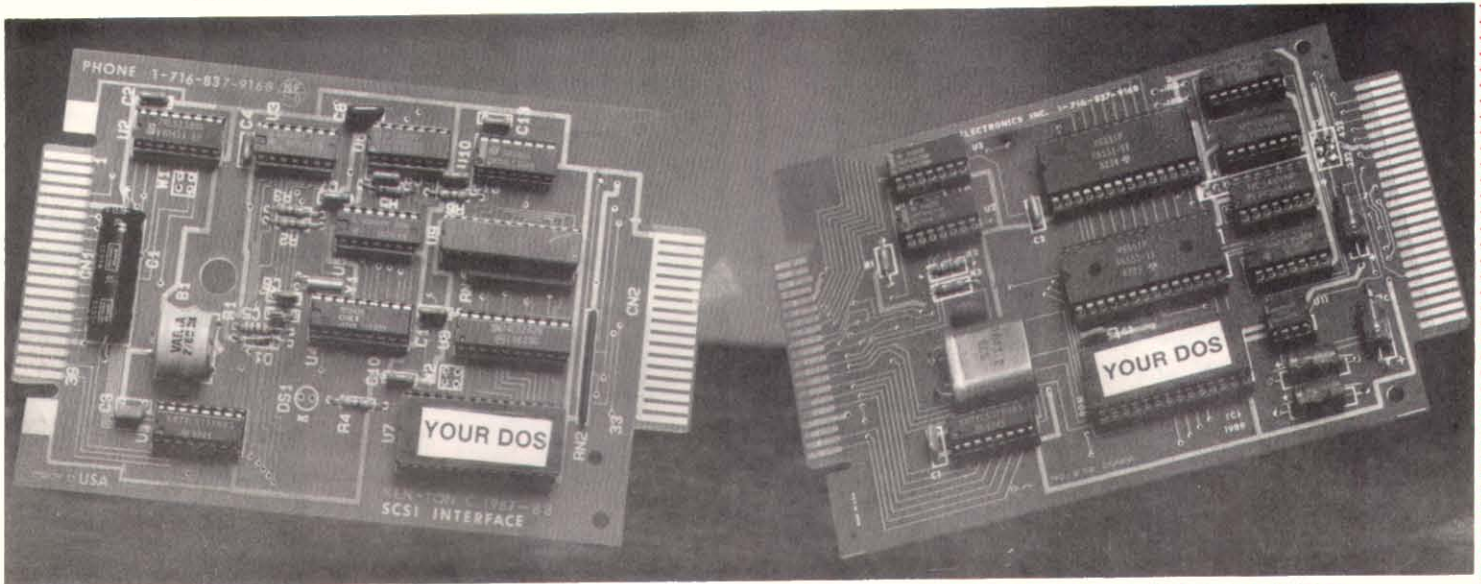
☎(503) 663-2865 ☐POB 1233 Gresham, OR 97030
Non VIP Library orders add \$3 for shipping and handling in USA. Canada \$4. Foreign \$6. COD orders add an additional \$2.25. Checks allow 3 weeks for delivery. All other orders are shipped the same day.

Writer III or Library /W owners: Upgrade to the VIP Writer III 2.0 for \$10 + \$3 S/H. Send ORIGINAL disk and \$13 total.

Telewriter 128 is a trademark of Cognitec. Word Power 3 is a trademark of Microcom Software.

KEN-TON ELECTRONICS PRESENTS

"REAL" S.C.S.I. INTERFACE -AND- THE DUAL RS-232 PAK



HARD DRIVE INTERFACE \$89 or \$119(with RTC)

- Real Time Clock Battery-backed
- L.R. Tech Compatible
- Owl Dos Compatible
- RGB Dos Compatible
- H-Dos Compatible
- OS-9 Compatible
- 28 pin ROM Socket

DUAL COMM BOARD \$74(Single) \$89(Dual)

- Replaces RS-232 PAK
- 2 - 6551 A.C.I.A.'s
- 2 Independent RS-232 Channels
- Jumper selectable for up to 4 (FOUR) Channels (with 2nd board)
- Ultra low power draw
- 28 pin ROM Socket

Build your Hard Drive the RIGHT way with a REAL S.C.S.I. Interface. All our products are MIL-Specification Quality P.C. Boards and carry a full 90 day Warranty. Both the Dual Comm and the SCSI Interface work directly with a Y-CABLE or the Multi-Pak Interface and are Made in the U.S.A.

CALL US FOR PRICES ON CUSTOM SYSTEMS, HARD DRIVES AND CABLES

Terms:
Check or M.O. accepted (US Funds only)
Please add \$4.00 for S&H
Phone Orders are welcomed!
CALL 1-716-837-9168 (24 hr. order line)

 **KEN-TON
ELECTRONICS**
187 GREEN ACRES RD.
TONAWANDA, NY 14150

**CoCo
Clipboard Magazine**

3742 U.S. 20 Box 3, • Fredonia, N.Y. 14063

BULK MAIL
U.S. POSTAGE
PAID
PERMIT NO. 18
FREDONIA, N.Y.
14063