The PORTABLE Color Computer!

Put your CoCo in almost ANY case... it isn't that hard!
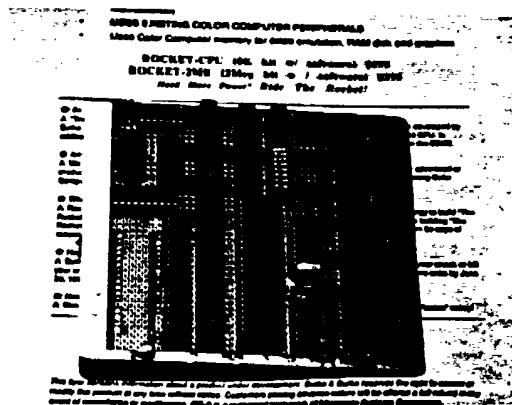
14MHz 16 bit "ROCKET" internal processor board
by Burke & Burke

The editor speaks...

Welcome subscribers! I wish to take this opportunity to thank you all for subscribing, and to reassure you that this magazine will continue to be published for a MINIMUM of two years. That is my personal commitment to you. If you enjoy the content enough to continue subscribing, then of course it will stay around much longer, something that I would very much like! We are currently growing steadily at around 20 new subscribers per month. Note that these are PAID IN FULL subscriptions, not just names on a free mailing list. If this rate continues, we will be publishing for quite a while! Should the time come that publication must cease, I WILL let you know several issues in advance. I do not like getting an unexpected last issue, and don't intend to give one if reasonably avoidable. In such a case, subscriptions will either be continued out or refunds issued.

The following is the tentative publication schedule:
01 August
15 September
01 November
15 December
01 February
15 March
01 May
15 June
Subscribers should receive their magazine within seven days before or after the above dates (i.e.- this issue should have been received between 26 July and 07 August). I will let you know of any changes in this column.

A lot has happened in the CoCo community since "the Rainbow" ceased publication in May. There have been THREE CoCoFests. Several new products have been introduced. You will find news concerning the 'Fests in the very first article. New products are discussed in "micro News".

If you have never attended a 'Fests, don't forget that the annual Atlanta CoCoFest is tentatively scheduled for the first weekend in October. Now is a good time to make plans to attend!

Several new authors appear in this issue as well as a few familiar names. If you have something you wish to share with the community, please drop us a note in the mail or call one weekend. You don't have to be a first rate author to have something printed! Just explain your article as best you can in writing. If there is something I don't understand, I'll give you a call! It is that simple, but you *must* make that first contact! "68' micros" can always use interesting programs, hardware modifications, and general articles. And I desire your criticisms and suggestions for improvements also! After all, this is YOUR magazine!

"the world of 68' micros" is not the first magazine to combine 6800 series and 68000 series Motorola processor information under one cover. That honor belongs to the long defunct "68 Micro Journal", often referred to as "68 Micro". 268'm (my pet abbreviation... it is actually a play on i286...) is not a reincarnation of "68 Micro". The name was not chosen because of a resemblance either. We can only hope to become as well known and respected as "68 Micros" has been.

Remember, 268'm is not limited to any single platform! If you use OS-9 on an Atari ST or Macintosh, one of the FHL/IMS/PT machines, or the CoCo, you have a reason to subscribe. We support ANY programming language that runs under OS-9 or the CoCo's native BÁSIC (disk or tape!). Programs are accepted for all! So what are you waiting for... an invitation to write? Well, this is it! Note that a "letters" column will begin in the next issue, so let us know how we are doing.

Please enjoy this premeire issue. Pass it around and encourage your friends to subscribe also. Send in your comments and articles. And when the time comes, I hope you will find that the magazine has proven interesting enough to continue your subscription...

Francis G. Swygert

A Great Year for CoCoFests!

F.G. Swygert

There will be no less than FOUR CoCoFests this year! Three have already passed (reviews follow) while one is yet to be... so you still have time to get in on the fun!

Let's start this article with the chance to attend a 'Fest this year... **The Fourth Annual Atlanta CoCoFest (02-03 October, 1993).** The Atlanta Computer Society, the oldest computer club in Atlanta (and possibly the U.S.A.!), has been hosting a 'Fest for the past three years. The location is the Northlake Holiday Inn, Exit 28 of I-285 (LaVista Road/Hwy 236). Head toward downtown Atlanta from the Interstate and the Holiday Inn entrance will be to the right. There are several other motels and many fast-food restaurants in the area, as well as Northlake Mall. Last year's 'Fest was great, with around 200 attendees and 20 vendors. This year should be at least as good, if not better! Rooms will be $52 (CoCoFest rate), booths $25. For more information contact ACS, c/o Alan Dages, 4290 Bells Ferry Road Suite. 10639, Kennesaw, GA 30144; phone 404-469-5111; club BBS 404-636-2991.

Three CoCoFests have already transpired this year. Even when Rainbow Magazine was sponsoring the 'Fests, there were not this many in one year!

The most recent 'Fest was **the Third Annual Pacific North West (PNW) CoCoFest** held June 25-26. This 'Fest is sponsored by the Port O' CoCo club of Port Orchard, Washington (south west of Seattle, across Puget Sound; just south of Bremerton).

In an attempt to increase attendance, other computer types were invited to attend. The show was predominantly CoCo (about 60%), but representatives with Apple, Commodore, and PC computers had displays also. This doesn't sound like a bad idea... maybe an "orphan" computer show would draw a much bigger crowd and make the show more worthwhile for the sponsors.

The PNW 'Fest has always been a small one. Approximately 75-80 people showed up to view the new products. Though only a few vendors were present, many products were represented. Andre LaVelle of SBUG had made arrangements at the Chicago fest to represent vendors who wouldn't be able to attend (such as myself) due to the distance. Andre was given a small percentage of sales, but his service was much more valuable than any remuneration he received, both for the vendors represented and the users who may not have seen the products.

The items that received the most attention were **Burke & Burke's "Rocket"** prototype (first seen in Chicago) and **"Power Boost"** OS-9 Level 2 upgrade for the 6309, and also **Gale Force's** rival OS-9 Level 2/6309 upgrade, **"NitrOS9"**. The Gale Force upgrade is the lesser known, but is more extensive the B&B upgrade. Gale Force disassembled *nearly every module* of OS-9 *by hand* and then reassembled with modifications taking advantage of the 6309's capabilities. No accurate comparisons have been done between the two to this date.

The following vendors were able to attend the PNW 'Fest:
Burke & Burke
Atlanta Computer Society
Gale Force
Bob van der Poel
South Bay Users Group (SBUG)

It is hoped that the organizers of this event might be able to move it across Puget Sound in the greater Seattle area next year. This would probably increase attendance. Personally, I wouldn't mind sharing the floor with some of the other "orphan" computers also... sounds like a very good idea, and a way to get some more assistance from other user groups. We orphans need to stick together also!

For information on next year's event, contact Port O' CoCo, c/o Donald Zimmerman, 3046 Banner Road, Port Orchard, WA 98366-8810; phone 206-871-6535

The Second Annual "Last" Chicago CoCoFest was held May 1-3 at the Holiday Inn of Elgin, just west of Chicago. The name comes from the fact that the year before (1992) CoCoPRO! sponsored it's last 'Fest in Chicago. Well, the Glenside CoCo Club (the co-sponsor of 1992, sponsor for 1993) sort of picked up on the name as a joke and continues to call each 'Fest the "last". There will be more to come, with next year already in planning!

This year there were a total of 30 vendors! Several clubs were represented in that total, among them the Overseas OS-9 User Group Consortium. This booth had representatives from Australia, Japan, Switzerland, and the Netherlands. There was a conference during the 'Fest concerning the future of OS-9. One item of interest that came up during this conference was the number of OS-9 users overseas. It has often been quoted that OS-9 was more popular overseas than in the U.S. Well, it turns out that this is not true! The Japanese, European, and U.S. user groups all have approximately 300 members each, while the Australian group is a bit smaller.

Several new items of interest were shown for the first time in Chicago: the Disto 2MB upgrade, Burke&Burke's "Rocket" 68000 CPU board for the CoCo, Sub-Etha's pull-down menu user interface for OS-9/OSK, CoNect's super high speed serial port, and the book "Tandy's Little Wonder" and a four page flyer for "the world of 68' micros" magazine from FARNA Systems (read more about these items in "Micro News").

Vendors Attending Chicago 'Fest:
Atlanta Computer Society
Adventure Survivors
BARSoft
Burke&Burke
Chicago Area OS-9 Users Group

ColorSystems
CoNect
Cook County CoCo Club
Dave & Nancy Myers
Delmar
Disk N' Dat
Disto
FARNA Systems
Frank Hogg Laboratories
Glenside CoCo Club
HawkSoft
Interactive Media Systems (IMS)
Kala Software
Dirt Cheap Computer Stuff Co.
OS-9 Users Group, U.S.A.
Overseas OS-9 User Group Consortium
Microware
Radical Electronics
South Bay Users Group (SBUG)
StG Net
StrongWare
Sub-Etha Software
Sundog Systems
T&D Software
Lee Veal

Contact Glenside CoCo Club, c/o Tony Podraza, 119 Adobe Circle, Carpentersville, IL 60110-1101; phone 708-428-3576 for information on the Third Annual "Last" Chicago CoCoFest. Carl Boll, a club officer, can also be contacted via Delphi, username CBJ; Internet cbj@delphi.com. The club supports a BBS at 708-428-0436.

The first 'Fest of the year was hosted by Mid Iowa & Country CoCo club. This is a national club with members all over the U.S. and Canada, as well as a few overseas. Des Moines, also home to Microware, was the scene of the March 27-28 event.

The "Middle America" 'Fest was rather small, with eleven vendors scheduled but only nine making it. Burke & Burke had made plans to attend, but had to cancel at the last minute, as did Zebra Systems. Part of the reason for the low attendance (roughly 75 attendees) was due to the close date of the Chicago 'Fest. Only 33 days and approximately 350 miles separated the two events. Many vendors and CoCo supporters

simply had to choose between the events. Those who had a long distance to travel chose the Chicago event, since it has long been THE CoCo event to attend. Hopefully more can be done to separate the events by at least two months, as the effort put out by MI&CCC was respectable. At least Terry Simons, president of the club, reports that the club didn't lose any money, and there should be an event planned for next year. Contact Terry at 1328 48th Street, Des Moines, IA 50311; phone 515-279-2576; Delphi username MRUPGRADE; Internet mrupgrade@delphi.com.

A neat attraction at Middle America was the presence of a couple of people from Microware- M. Hawkings and T. Earls. The names should sound familiar... they are two of the three people on the CoCo 3 CTRL-ALT-RESET screen! The third person left Microware after the CoCo 3 project to work for - ugh - Micrsoft. A DS-69B was set up and connected to a laser printer using the HP Ink Jet printer driver. One could have their picture taken and a disk file would later be sent (for $3) so that one could be the third person in their CoCo (after a ROM was burned)!

Vendors Attending Mid America 'Fest:
Atlanta Computer Society
Glenside CoCo Club/Chicago 'Fest
Mid Iowa & Country CoCo Club
South Bay Users Group (SBUG)
Sportsware, Inc.
StG Net
Sub-Etha Software
Sundog Systems
Dave Wordell

If you have never been to a CoCoFest, try to make plans to attend one! Not only is there lots of new and used hard and soft ware, but there are also many free, interesting seminars. If there isn't one in your area, maybe you could enlist the aid of a local club and have one. Any of the above hosts as well as the editor of this magazine would be willing to give advice and assistance. Do remember that there should be some time and distance between the 'Fests. A

California 'Fest at the same time as the Atlanta 'Fest shouldn't impact many attendees, but will impact some of the major vendors. Keep this in mind when planning! The two established 'Fests, Atlanta and Chicago, will most likely be the most attended by vendors for some time to come. Who knows, maybe I'll see you in Atlanta this year! Since Atlanta is practically in my back yard (about 150 miles north of Warner Robins), FARNA Systems will definitely be there!

Tips for planning a successful 'Fest:

* Plan far in advance! You need to start getting the word out to vendors and attendees at least six months before the planned date. This means a lot of foot work must be started almost a year before the planned date, especially site negotiations.
* Choose a site that is easily accessible to a large number of people. You will draw a lot of walk-ins that way, people who may normally stay home, as they aren't the CoCo die-hards that "we" are!
* Advertise in as many publications as possible, even local papers. You'd be surprised how many closet CoCo users there are! You might even entice someone to bring theirs back out...
* Consider inviting local Apple, Commodore, Texas Instruments, Adam, and other "orphan" makes to join you. It will make the event bigger and more attractive to a hotel to host. Personally, I'd leave the Macs and PCs out, as they tend to overshadow everything else...
* Keep costs down! Some of us travel great distances to attend a 'fest. A small 'Fest doesn't have to be at a Holiday Inn, a cheaper hotel will do nicely. An alternative is to put phone numbers of lower cost hotels in your advertising.
* The best advice is to call or write the organizers of the other events. All have made mistakes at some point and should be able to offer some wisdom... better for you to learn from their mistakes rather than your own!
* Finally, let me know! I will happily print any news of upcoming events and assist in any way possible.

RGB Color Display ...

Richard S. Bair

A Complete 64 Color Onscreen Chart for the CoCo 3 with RGB monitor

When I purchased an RGB monitor for my CoCo 3 I felt that I had finally "arrived". 80 columns, and no herringbone interference patterns across the screen! But two disappointments awaited: First, I discovered that lots of the Tandy software was now in boring black-and-white instead of delightful colors. Second, I couldn't make any sense out of the relationship between the 64 RGB colors and their numbers.

The cause of the first problem has since been explained to me, and I am resigned to it. For the second, a solution came from the Tandy Service Manual for the CoCo 3, which explains how the different bits of the color number specify the strength of the red, green, and blue signals. I was pleased at the knowledge that there is a pattern to the colors, but this led me immediately to a challenge: Could a color chart be devised that would make the pattern obvious?

Easier said than done. The TV/composite color set is basically two dimensional, so it's easily displayed in an organized chart. The RGB color set, on the other hand, is three-dimensional; a truly correct RGB color chart would resemble a 4x4x4 Rubik's cube with every little cube a different color, including ones buried inside!

It is tempting to consider simply slicing the "RGB cube" into 4 layers which could then be 2-dimensional charts, but doing so obscures the logic of the colors; they are not in groups of 16 at all. Insofar as the cube has any symmetry, it is around an axis that runs from one corner (black,0) to the diagonally opposite corner (white,63). So the best 2-dimensional reduction turns out to be an extraction from the cube of a series of "rings" of colors that circle the axis.

RGBCLRDS displays the colors in three groups. The top group contains the three primary colors in their three available levels of brightness (plus black), and also all the colors created

by mixing just two of the primaries in various proportions. I call these secondary colors, though my dictionary says this is not the correct use of the phrase. Lower on the screen are the tertiary colors, which mix all three of the primaries in various proportions. At the bottom are the two "gray-scale" colors which lie on the axis of the "RGB cube" between black and white.

Within each group, the colors are arranged in order of brightness from left to right, and each vertical band represents one of the "rings" mentioned above, cut midway between red and blue and flattened out. If you study the display for a while, you will see that the progression of colors in each column leads from the bottom color to the one on top again.

The BASIC program "RGBCLRDS" (RGB Color Display) first loads an ML program called "PALTJUGL" (Palette Juggle), then sets to work drawing the various color boxes on the screen. Most of this drawing is invisible, because all palette registers but one are set to the same color. Each color box is assigned a palette register number, but of course since there are 64 colors and only 16 registers, they begin to repeat after a while. When the BASIC program finishes drawing it turns control over to PALTJUGL, and the 64 colors appear. At this point you can have some fun playing with the brightness and contrast controls to see how they affect the different colors. Several of the darker ones are totally black at low brightness levels.

PALTJUGL accomplishes a 64 color display by continually changing the contents of the palette registers in synch with the display scan, so that each register contains the appropriate color at every point in the display. Even though nothing moves on the screen at this point, the CPU is busy! To exit the display, use CTRL-ALT-RESET, RESET. A simple RESET doesn't work. (Leaves the stack corrupted, I guess.)

RGBCLRDS will work on cassette or RS floppy systems, but may not work right on a machine with any non-standard ROM (ADOS users may want to disable ADOCS first). PALTJUGL is sensitive to the length of the IRQ routine because palette accesses must occur while the display trace is offscreen, otherwise streaks appear on the display. The only change between cassette and disk versions is in line 30 of the BASIC program. "PLJGLMKR" (Palette Juggle Maker) creates PALTJUGL from DATA lines. If you plan to save them to cassette, record PALTJUGL immediately after RGBCLRDS on the tape. Have a colorful day!

```
10 REM "RGBCLRDS" Color display
for CoCo 3 RGB monitor
15 (c) 1993 FARNA Systems PB, written
by Steven S. Bair
20 CLEAR 200, &H7EFF
30 LOADM "PALTJUGL":REM For tape
system change this step to CLOADM
"PALTJUGL"
40 FOR PR=0 TO 15
50 PALETTE PR,57
60 NEXT PR
70 PALETTE 1,1
80 HSCREEN 2
90 HPRINT(0,0),"The Color Computer
RGB Monitor Color Set"
100 HLINE(0,7)-(319,7),PSET
110 HPRINT(6,1),"<Darker
Brighter>"
120 HLINE(0,15)-(319,15),PSET
130 HPRINT(14,21),"Please wait."
140 FOR C=0 TO 63
150 READ PR,X1,Y1,X2,Y2
160 HCOLOR PR,0
170 HLINE(X1,Y1)-(X2,Y2),PSET,BF
180 IF X1>159 THEN HCOLOR 1,PR
ELSE HCOLOR 0,PR
190 IF Y1/8>INT(Y1/8) THEN LET
Y1=Y1+4
200 LET Y3=(Y1/8+INT((Y2-7)/8))/2
210 LET C$=STR$(C)
220 IF LEN(C$)=2 THEN LET
C$="0"+RIGHT$(C$,1)ELSE LET
C$=RIGHT$(C$,2)
230 HPRINT(X1/8+1,Y3),C$
240 NEXT C
250 HCOLOR 1,0
260 HPRINT(6,2),"Primary and Secondary
Colors"
```

270 HLINE(48,23)-(271,23),PSET
280 HPRINT(12,13),"Tertiary Colors"
290 HLINE(96,111)-(215,111),PSET
300 HLINE(112,168)-(207,175),PRESET,BF
310 HPRINT(4,21),"Black > > Gray > > White"
320 HLINE(32,175)-(287,175),PSET
330 HPRINT(6,23),"To exit, press RESET button."
340 EXEC
350 DATA 2,0,24,31,183,1,32,24,63,47,14,32,48,63,71,10,64,40,95,51,13,32,72,63,95,12,64,88,95,99,10,64,64,95,75,3,88,176,135,183,3,64,28,95,39,7,96,32,127,39,11,96,40,127,47,8,128,32,159,39,4,96,24,127,31,5,128,24,159,31,6,128,116,159,131
360 DATA 7,160,120,191,127,5,64,52,95,63,15,96,48,127,55,6,96,56,127,63,3,128,48,159,55,11,96,64,127,71,15,128,132,159,147,7,128,56,159,63,3,160,136,191,143,12,128,40,159,47,13,160,36,191,47,4,160,48,191,59,9,192,32,223,55,12,160,128,191,135
370 DATA 9,192,120,223,127,13,192,128,223,135,14,224,128,255,135,5,64,76,95,87,14,96,88,127,95,15,96,72,127,79,13,128,148,159,163,6,96,80,127,87,7,128,80,159,87,3,128,72,159,79,14,160,152,191,159,15,128,88,159,95,6,160,24,191,35,8,160,112,191,119
380 DATA 10,192,112,223,119,11,160,84,191,95,9,192,80,223,103,9,192,152,223,159,4,224,112,255,119,12,128
,64,159,71,10,160,144,191,151,9,160,60,191,71,4,192,136,223,143,4,160,72,191,83,6,192,144,223,151,8,192,56,223,79,12,224,144,255,151
390 DATA 4,184,176,232,183,0,224,120,255,127,7,224,136,255,143,11,256,124,287,139,11,224,152,255,159,3,256,108,287,123,8,256,140,287,155,5,288,112,319,183

10 REM "PLJGLMKR" PALTJUGL MAKER
20 CLEAR 200,&H7EFF
30 WIDTH 40
40 FOR X=0 TO 14
50 T=0
60 FOR Y=0 TO 15
70 READ A$
80 A=VAL("&H"+A$)
90 POKE &H7F00+16*X+Y,A
100 T=T+A
110 NEXT Y
120 READ A$
130 IF VAL("&H"+A$)<>T THEN GOTO 400
140 NEXT X
150 PRINT "ML program is now in memory; to save totape, GOTO 200; to save to disk,   GOTO 300."
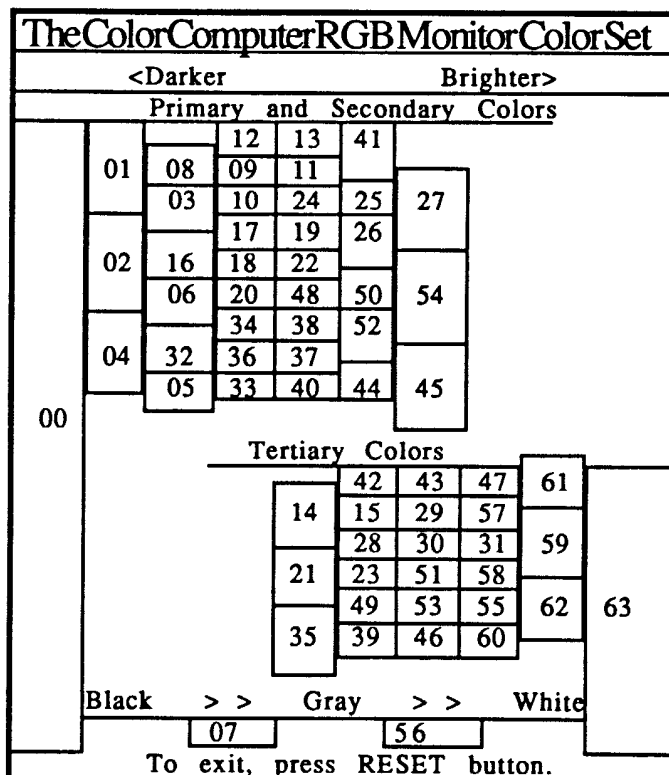160 END
200 CSAVEM "PALTJUGL",&H7F00,&H7FEC,&H7F00
210 END

300 SAVEM "PALTJUGL",&H7F00,&H7FEC,&H7F00
310 END
400 PRINT "ABORT: DATA ERROR IN LINE";:PRINT 1000+10*X
410 END
1000 DATA 7F,FF,D8,7F,01,13,B6,01,13,27,FB,CE,7F,61,86,FF,808
1010 DATA D6,BC,C1,0E,27,06,12,12,12,12,12,12,12,12,12,12,342
1020 DATA 12,37,04,10,8E,00,02,31,3F,26,FC,5A,1A,00,12,12,317
1030 DATA 10,8E,00,06,31,3F,26,FC,5A,26,F5,37,04,F7,7F,60,5BC
1040 DATA 37,04,1F,02,37,04,E7,A4,10,8E,00,02,31,3F,26,FC,454
1050 DATA 1A,00,12,7A,7F,60,26,E8,11,83,7F,ED,27,A5,20,C1,640
1060 DATA 00,02,0E,B2,00,B3,08,B4,0C,B5,0D,B6,29,B7,09,B8,556
1070 DATA 0B,B9,1B,BA,03,BB,0A,BC,18,BD,19,BE,02,BF,11,40,5DB
1080 DATA 06,B3,13,B4,1A,B5,10,B6,12,B7,16,B8,36,0C,0A,B9,5B1
1090 DATA 32,BA,06,BB,14,BC,30,BD,04,BF,22,B3,26,B4,34,B5,6C5
1100 DATA 20,B6,24,04,04,B7,25,B9,2D,BB,2C,BC,05,04,04,BE,532
1110 DATA 21,BF,28,B3,3D,B8,2A,14,0A,BA,2B,B4,2F,B5,3F,B6,66A
1120 DATA 0E,B7,0F,B9,1D,BB,3B,BC,1C,BD,1E,BE,1F,10,0A,BF,609
1130 DATA 15,B3,17,B4,33,B7,3A,B8,3E,BA,31,B6,35,BC,37,BD,733
1140 DATA 23,BE,27,04,04,B9,2E,BB,3C,B3,07,B4,38,00,00,00,494

The aspect ratio is off, but this is a good representation of the display "RGBCLRDS" generates. It is much better in full color!

NOTE: The assembly code listing for PALTJUGL is on "microdisk"

The Color Computer RGB Monitor Color Set

<Darker   Brighter>

Primary and Secondary Colors

00	01	08	12	13	41	
		08	09	11		
		03	10	24	25	27
	02	16	17	19	26	
		16	18	22		
		06	20	48	50	54
	04	32	34	38	52	
		32	36	37		
		05	33	40	44	45

Tertiary Colors

14	42	43	47	61	
14	15	29	57		
21	28	30	31	59	
21	23	51	58		
35	49	53	55	62	63
35	39	46	60		

Black > > Gray   > > White

07	56

To exit, press RESET button.

Little Black Book

Jason Reighard

A versatile address, telephone number, and mailing label database.

Little Black Book (LBB) is a program to conveniently store addresses and phone numbers. You can use it to print mailing lists, mailing labels, and phone number lists. It is completely menu driven and easy to use. LBB will run on any Disk Extended BASIC Coco with at least 32K. The program checks for the presence of a Coco 3 and will take advantage of some of it's features, but will run unchanged on any CoCo.

The program automaticly sets your disk drives to run at 6 ms. If you get an I/O error, simply go to the drive options and select a slower rate. LBB also has an alphabetizing feature that comes in very handy.The entire database is stored in memory to make sorting and searching lightning fast. If the memory limit is reached, simply save your existing data as a file and start another. There is no limit to the number of files except disk space.

Note that line 25 reads "PORTIONS BY L.TODD KNUDSEN". Mr. Knudsen contributed the code for error free double speed disk access (lines 80-140). As long as line 25 is included, the code may be used in any program. This information was originally published in the Glenside CoCo Club newsletter (CoCo 123).

If there are any questions or suggestions, the author can be contacted at: 441 Ridgeland Dr., Toronto, Ohio 43964-2022; phone (614)-537-4875. Please include a S.A.S.E if requesting a reply. Upgrades and modifications should be sent to the publisher for future article considerations. Anyone care to convert this program to Basic-09?

Jason will be a Senior at Edison High School (94-95 school year) and has used a Coco since second grade. He also likes electronics and ham radio.

```
10 ' LITTLE BLACK BOOK, BY JASON REIGHARD, 1992,1993
20 ' COPYRIGHT 1993, FARNA SYSTEMS
25 ' PORTIONS BY L.TODD KNUDSEN
30 CLEAR 3000:DIMN$(300), T$(300), A$(300), S$(300)
40 IF PEEK(33021)=50 THEN WIDTH32 ELSE 70
50 POKE 65497,0
60 ON ERR GOTO2200
65 ' CHECK FOR ADOS/EADOS
70 IF PEEK(&HA282)=23 THEN 180 'ADOS OR EADOS PRESENT
75 ' DOUBLE SPEED FOR DISK DRIVES
80 IF PEEK(&HC004)=214 THEN A$="C 0EED52AD6D1D6F1D727D75E" ELSE A$="C101D617D7C4D7E4D 81A D 851"
'214=DECB 1.0, 215=1.1
90 FOR V=1TO24 STEP 4
100 A=VAL("&H"+MID$(A$,V,4))
110 POKE A,189:POKE A+1,240:POKE A+2,157:POKE A+3,18
120 NEXT
130 POKE      &HF09D,52:POKE &HF09E,127
140 POKE      &HF09F,53:POKE &HF0A0,255
150 IFPEEK(&HC004)=215THEN170
160             POKE&HD6CD,0: POKE&HD723,20:  GOTO180
170 POKE&HD7C0,0:POKE&HD816,20
175 ' BEGIN PROGRAM
180 DN=PEEK(235):N=0:CLS:PRINT" LITTLE BLACK BOOK 1.1"
190 PRINT"COPYRIGHT 1992,93 FARNA SYSTEMS"
200 PRINT:PRINT"   PRESS ANY KEY"
210 Z$="b":GOSUB 2130
220 Z$="bl":GOSUB 2130
230 Z$="bla":GOSUB 2130
240 Z$="blac":GOSUB 2130
250 Z$="black":GOSUB2130
260 Z$="blacK":GOSUB2130
270 Z$="blaCK":GOSUB2130
280 Z$="blACK":GOSUB2130
290 Z$="bLACK":GOSUB2130
300 Z$="BLACK":GOSUB2130
310 GOTO210
320 ' LOAD
330 CLS: PRINT"LOAD": PRINT@17 , "DEFAULT DRIVE"DN:PRINT"LITTLE BLACK BOOK FILES DRIVE" DN: GOSUB3130:LINEINPUT"FILENAME>"; F$: IF F$=""THEN 440 ELSE IF PEEK (33021) =50 THEN 360 ELSE340
340 IFLEN(F$)>8 THEN ER=31: GOTO
2200 ELSE FORX=1TO8: IF MID$ (F$,X,1)="/"OR MID$(F$,X,1)="."THEN ER=31 :GOTO2200
350 GOSUB3020:IF OD=0 THEN ER=26: GOTO2200
360 OPEN"I",#1,F$+"/LBB"
370 N=N+1
380 INPUT#1,N$(N),T$(N), A$(N), S$(N)
390 IF EOF(1)=-1 THEN 420
400 N=N+1
410 GOTO380
420 CLOSE#1
430 ' MAIN MENU
440 CLS
450 PRINT"*MENU*": PRINT@17, "DE-FAULT DRIVE"DN
460 PRINT"1 VIEW ENTRIES"
470 PRINT"2 EDIT ENTRIES"
480 PRINT"3 PRINT"
490 PRINT"4 LOAD"
500 PRINT"5 SAVE"
510 PRINT"6 DISK OPTIONS"
520 PRINT"7 ERASE ALL ENTRIES"
530 PRINT"8 ALPHABETIZE"
540 PRINT"9 END"
550 INPUT"SELECT1-9";M$
560 M=VAL(M$)
570 IFM<1ORM>9THEN440
580 ON M GOSUB 610, 1010, 1470, 330, 2030, 2340, 2650, 2680, 2300
590 GOTO440
600 ' VIEW MENU
610 CLS: PRINT"VIEW ENTRIES": PRINT
620 PRINT"1 VIEW ALL ENTRIES"
630 PRINT"2 SEARCH FOR A ENTRY"
640 PRINT"3 RETURN TO MAIN MENU"
650 INPUT"SELECT(1-3)";M$
660 M=VAL(M$)
670 IFM<1ORM>3THEN610
680 ON M GOTO700,850,690
690 RETURN
700 FORX=1TON
710 CLS
720 PRINTN$(X)
730 PRINTA$(X)
740 PRINTS$(X)
750 PRINTT$(X)
760 PRINT
770 PRINT"PRESS   ENTER   TO CONTINUE":PRINT"PRESS X TO EXIT"
780 A$=INKEY$
790 IFA$=CHR$(13)THEN 820
800 IFA$="X"THEN 610
810 GOTO780
820 NEXTX
830 GOTO610
840 ' SEARCH
850 CLS:LINE INPUT"NAME TO
```

```
SEARCH FOR          >>> ";M$
860 IF M$=""THEN 610
870 Q=LEN(M$)
880 FORX=1TON
890 IF M$ = LEFT$(N$(X),Q) THEN 900
ELSE 970
900 CLS
910 PRINTN$(X)
920 PRINTA$(X)
930 PRINTS$(X)
940 PRINTT$(X)
950 PRINT:PRINT"HIT ANY KEY TO
CONTINUE"
960 IF INKEY$=""THEN960
970 NEXTX
980 CLS
990 PRINT@0,"NOTHING MORE UN-
DER THAT ENTRY"; :PRINT@32,
"PRESS ANY KEY";: A$=INKEY$:
IFA$=""THEN 990 ELSE610
1000 ' EDIT
1010 CLS:DE$="":PRINT"EDIT EN-
TRIES"
1020 PRINT
1030 PRINT"1 CHANGE ENTRY"
1040 PRINT"2 ADD ENTRIES"
1050 PRINT"3 DELETE ENTRY"
1060 PRINT"4 RETURN TO MAIN MENU
1070 INPUT"SELECT(1-3)";M$
1080 M=VAL(M$)
1090 IFM<1ORM>5THEN1010
1100 ON M GOTO1130,1350,1190,1110
1110 RETURN
1120 ' CHANGE
1130 CLS
1140 IA$="      CHANGE": AA$="
CHANGE": A$="  CHANGE": GOSUB
2810
1150 IFZ$="N"THEN 1010
1160 IF DE$="N"THEN1010
1170 IFM$=""THEN 1010
1180 GOTO1260
1190 IA$="    DELETE": A$="ENTER
NAME YOU WISH TO DELETE":AA$="
DELETE":GOSUB2810
1200 IFDE$="N" THEN1010
1210 IFZ$="N"THEN1010
1220 IFM$="" THEN 1010
1230 INPUT"ARE YOU SURE(Y/N)??";
M$: IFM$="Y"THEN1240ELSE1010
1240 N=N-1:FORXX=XTON:N$(XX)=N$
(XX+1):T$(XX)=T$(XX+1): A$(XX)=A$
(XX+1): S$(XX)=S$(XX+1):NEXTXX
1250 GOTO1010
1260 M=X:CLS
1270 PRINTN$(M): PRINTT$(M): PRINT
A$(M): PRINTS$(M): PRINT STRING$
(32,"*");
1280 PRINT"PRESS [ENTER] FOR NO
CHANGE"
1290 LINEINPUT"NAME >>> ";M$:
IFM$<>""THENN$(M)=M$
1300 LINEINPUT"NUMBER  >>>

";M$:IFM$<>""THENT$(M)=M$
1310 LINEINPUT"ADDRESS >>> ";M$:
IFM$<>""THENA$(M)=M$
1320 LINEINPUT "CITY, STATE, ZIP
CODE >>> ";M$: IF M$<>"" THEN S$
(M)=M$
1330 GOTO1010
1340 ' ADD RECORDS
1350 CLS:PRINT"ADD RECORDS"
1360 N=N+1:PRINT"REC."N
1370 IF N>300 THEN 2640
1380 PRINT"TYPE '*END' WHEN YOU
DO NOT WISH TO ENTER ANY MORE
RECORDS"
1390 LINEINPUT"NAME >>> ";N$(N)
1400 IFN$(N)="*END"THENN=N-1:RE-
TURN
1410 LINEINPUT"NUMBER >>> ";T$(N)
1420 LINEINPUT"ADDRESS >>>
";A$(N)
1430 LINEINPUT"CITY, STATE, ZIP
CODE         >>";S$(N)
1440 IF N>=300THENN=300:RETURN
1450 GOTO1350
1460 ' PRINT ENTRIES
1470 CLS:PRINT"PRINT"
1480 PRINT:PRINT"1 PRINT ALL EN-
TRIES": PRINT"2 PRINT SINGLE EN-
TRY": PRINT"3 CHANGE BAUD RATE":
PRINT"4 RETURN TO MAIN MENU"
1490 LINEINPUT "SELECT 1-3 ";A$
1500 A=VAL(A$)
1510 IFA<1ORA>5 THEN1470
1520 ON A GOTO1540,1710,1900,440
1530 GOTO1470
1540 CLS:PRINT"PRINT ALL ENTRIES":
PRINT
1550 PRINT"OPTIONS":PRINT"1 PRINT
NAME, ADDRESS AND NUMBER";:
PRINT"2 PRINT NAME AND NUMBER
ONLY":PRINT"3 PRINT NAME AND AD-
DRESS ONLY": PRINT"4 ABORT"
1560 LINEINPUT"SELECT 1-4 ";Z$
1570 IFZ$="4" THEN 1470
1580 Z=VAL(Z$)
1590 IF Z>4 THEN1540
1600 IF Z<1 THEN1540
1610 LINEINPUT"HOW MANY LINES
TO SKIP BETWEEN LABELS ";LL$:
LINEINPUT"MARGIN";MM$:MM=VAL
(MM$)
1620 IF LL$=""THEN 1470
1630 LL=VAL(LL$)
1640 CLS0
1650 FORX=1TO N
1660 PRINT#-2,TAB(MM);N$(X)
1670 IFZ=1ORZ=3 THEN PRINT#-
2,TAB(MM); A$(X)
1680 IFZ=1ORZ=3 THEN PRINT#-
2,TAB(MM); S$(X)
1690 IFZ=1ORZ=2 THEN PRINT#-
2,TAB(MM);T$(X)
1700 FOR  Q=1TO LL:PRINT#-2:

NEXTQ: NEXTX: GOTO1470
1710 CLS: IA$="PRINT": AA$="PRINT":
GOSUB 2810
1720 IFZ$="N"THEN1470
1730 IFDE$="N"THEN1470
1740 IF M$=""THEN1470
1750 CLS:PRINT"OPTIONS":PRINT"1
PRINT NAME, ADDRESS AND NUM-
BER";:PRINT"2 PRINT NAME AND NUM-
BER ONLY":PRINT"3 PRINT NAME AND
ADDRESS ONLY": PRINT"4 ABORT"
1760 LINEINPUT"SELECT 1-4 ";Z$:
Z=VAL(Z$): IFZ=4 THEN 1470 ELSE
LINEINPUT"MARGIN";MM$:MM=VAL
(MM$)
1770 IFZ$="4" THEN 1470
1780 IF Z<1 THEN1750
1790 IFZ>5 THEN1750
1800 CLS0:PRINT#-2,TAB(MM);N$(X)
1810 IFZ=1ORZ=3 THEN PRINT#-2,
TAB(MM);A$(X):PRINT#-2,TAB(MM);S$
(X)
1820 IFZ=2ORZ=1 THEN PRINT#-
2,TAB(MM);T$(X)
1830 CLS:PRINT"PRESS (SPACEBAR)
TO ADVANCE  PRINTER ONE LINE":
PRINT "PRESS (ENTER) TO PRINT
LABEL AGAIN": PRINT"PRESS @ TO
SELECT ANOTHER LABEL TO PRINT":
PRINT"PRESS M FOR MENU"
1840 A$=INKEY$
1850 IFA$=" "THEN PRINT#-2: GOTO
1830
1860 IFA$=CHR$(13)THEN 1750
1870 IFA$="@" THEN 1710
1880 IFA$="M" THEN 1470
1890 GOTO1840
1900 CLS:PRINT"CHANGE BAUD
RATE"
1910 PRINT:PRINT"1  600":PRINT"2
1200":PRINT"3    2400":PRINT"4
4800":PRINT"5 9600":PRINT"6  NO
CHANGE"
1920 LINEINPUT "SELECT 1-8 ";A$
1930 IFA$="1" THEN X=180:GOTO2000
1940 IFA$="2" THEN X=87:GOTO2000
1950 IFA$="3" THEN X=41:GOTO2000
1960 IFA$="4" THEN X=18:GOTO2000
1970 IFA$="5" THEN X=4:GOTO2000
1980 IFA$="6" THEN 1470
1990 GOTO1900
2000 POKE 150,X
2010 GOTO1470
2020 ' SAVE
2030 CLS: PRINT"SAVE": PRINT@17,
"DEFAULT DRIVE"DN: LINEINPUT
"SAVE FILENAME>>";F$:IFF$=""THEN
440 ELSE IF PEEK(33021)=50 THEN
2050  ELSE2040
2040 IF LEN(F$)>8 THEN ER=31 :
GOTO 2200 ELSE NEXTX: RETURN
2050 GOSUB3020:IF OD=1 THEN
PRINT "file already exists": PRINT
```

```
"OVERWRITE (Y/N)" ELSE 2080
2060 A$=INKEY$:IF A$="Y"THEN 2080
ELSE IF A$="N" THEN 440
2070 GOTO 2060
2080 OPEN"O",#1,F$+".LBB"
2090 FORX=1TON
2100 WRITE#1,N$(X),T$(X),A$(X),S$(X)
2110 NEXTX:CLOSE#1
2120 GOTO440
2130 FORX=1TO20
2140 A$=INKEY$
2150 PRINT@13,Z$;
2160 IFA$=""THEN NEXTX:RETURN
2170 GOTO440
2180 LL=VAL(LL$)
2190 ' ERROR MESSAGES
2200 CLS:PRINT"THE FOLLOWING
ERROR HAS OCCURED": PRINT:
SOUND1,3
2210 IF ERNO=30 THEN PRINT"THIS
DISK IS WRITE PROTECTED":
PRINT"CHECK DISK": GOTO2270
2220 IF ERNO=20 THEN PRINT"I/O
ERROR":PRINT"CHECK DISK DRIVE":
GOTO2270
2230 IF ERNO=31 THEN PRINT"BAD
FILENAME":PRINT"YOUR FILENAME
CANNOT BE MORE    THAN EIGHT
CHARACTERS LONG AND CANNOT
CONTAIN A . OR A / ":GOTO2270
2240 IF ERNO=26 THEN PRINT"FILE
NOT FOUND":GOTO2270
2250 IF ERNO=28 THEN PRINT"DISK
FULL ":GOTO2270
2260 IFERNO=6 THEN PRINT"OUT OF
MEMORY ERROR":GOTO2270 ELSE
PRINT"FATAL ERROR        #"ERNO"
IN LINE"ERLIN"
2270 PRINT:PRINT"PRESS ANY KEY
FOR MENU"
2280 A$=INKEY$: PRINT@14, "ER-
ROR";: IFA$="" THEN2290 ELSE 440
2290 A$=INKEY$: PRINT@14, "error";:
IF A$="" THEN2280 ELSE 440
2300 SOUND100,10: CLS: LINEINPUT
"ARE YOU SURE YOU WANT TO QUIT?
(Y/N)";A$
2310 IFA$="Y"THEN POKE65496,0:
POKE65495,0: END
2320 GOTO440
2330 'DISK OPTIONS
2340 CLS:PRINT@0,"DISK OPTIONS":
PRINT@32," 1 CHANGE DEFAULT
DRIVE"
2350 PRINT" 2 CHANGE STEP RATE"
2360 PRINT" 3 RETURN TO MENU"
2370 INPUT D
2380 IF D <1 OR D>3 THEN 2340
2390 IF D=1 THEN 2420
2400 IF D=2 THEN 2490
2410 IF D=3 THEN 440
2420 PRINT@17,"DEFAULT DRIVE"
DN;: PRINT"CHANGE DEFAULT

DRIVE": PRINT:PRINT"SELECT 0-
3":PRINT"PRESS [ENTER] FOR NO
CHANGE":PRINT:PRINT"CURRENT
DRIVE NUMBER "DN: PRINT: PRINT:
PRINT: PRINT
2430 A$=INKEY$:IFA$=""THEN2420
2440 IF A$=CHR$(13)  THEN 440
2450 IF VAL(A$)>5 THEN 2420
2460 DN=VAL(A$)
2470 IF DN>3 THEN 2420
2480 DRIVE DN:GOTO440
2490 CLS: PRINT@32,"SELECT STEP
RATE"
2500 PRINT" 1 = 6ms       2 = 12ms"
2510 PRINT" 3 = 20ms      4 = 30ms"
2520 PRINT:INPUT"Selected Value";D$
2530 IFPEEK(&HC004)=215 THEN 2540
ELSE 2580
2540 IFD$="1"THENPOKE&HD7C0,0:
POKE&HD816,20:  GOTO440
2550 IFD$="2"THENPOKE&HD7C0,0:
POKE&HD816,21:GOTO440
2560 IFD$="3"THENPOKE&HD7C0,0:
POKE&HD816,22:GOTO440
2570 IFD$="4"THENPOKE&HD7C0,0:
POKE&HD816,23:GOTO440
2580 IFD$="1"THENPOKE&HD6CD,0:
POKE&HD723,20:GOTO440
2590 IFD$="2"THENPOKE&HD6CD,0:
POKE&HD723,21:GOTO440
2600 IFD$="3"THENPOKE&HD6CD,0:
POKE&HD723,22:GOTO440
2610 IFD$="4"THENPOKE&HD6CD,0:
POKE&HD723,23:GOTO440
2620 GOTO 2490
2630 ' FULL
2640 CLS:PRINT"RECORDS FULL !!!!":
N=300: SOUND100,10: GOTO 440
2650 SOUND100,10: PRINT"ARE YOU
SURE??? (Y/N)";:INPUTA$:IFA$="Y"
THEN 2660 ELSE 440
2660 Q$="":FORX=1TON:N$(X)=Q$:
T$(X)=Q$:A$(X)=Q$:S$(X)=Q$:NEXTX:
N=0: GOTO440
2670 ' SELECTION SORT
2680 CLS0
2690 FOR Z= N TO 1 STEP -1
2700 MX$ = N$(1):MX = 1
2710 FOR ZZ = 1 TO Z
2720 IF N$(ZZ) > MX$ THEN MX$=N$
(ZZ): MX = ZZ
2730 NEXT ZZ
2740 TM$=N$(Z): N$(Z)=MX$: N$(MX)=
TM$
2750 TM$=T$(Z):T$(Z)=T$(MX): T$(MX)
=TM$
2760 TM$=A$(Z): A$(Z)=A$(MX): A$(MX)
=TM$
2770 TM$=S$(Z): S$(Z)=S$(MX): S$(MX)
=TM$
2780 NEXT Z
2790 RETURN
2800 ' LOCATE ROUTINE

2810 CLS:PRINT"ENTER THE NAME
YOU WISH TO"lA$;:LINEINPUT">>>";
M$
2820 IF M$=""THEN RETURN
2830 Q=LEN(M$)
2840 FORX=1TON
2850 IF M$ = LEFT$(N$(X),Q) THEN
2860 ELSE 2960
2860 CLS
2870 PRINTN$(X)
2880 PRINTA$(X)
2890 PRINTS$(X)
2900 PRINTT$(X)
2910 PRINT:PRINT"IS THIS THE EN-
TRY YOU WISH TO "A$" (Y/N)?"
2920 Z$=INKEY$
2930 IFZ$="Y"THEN RETURN
2940 IFZ$="N"THEN 2960
2950 GOTO2920
2960 NEXTX
2970 DE$="N"
2980 CLS
2990 PRINT@0,"NOTHING MORE UN-
DER THAT ENTRY";: PRINT@32,
"PRESS ANY KEY";: A$=INKEY$:IFA$=
"" THEN 2990 ELSE RETURN
3000 POKE 65496,0
3010 'FILE LOCATE
3020 OD=0:SC=3:F1$=F$
3030 IF LEN(F1$)<8 THEN FORX=1 TO
(8-LEN(F1$)):F1$=F1$+" ":NEXTX
3040 F1$=F1$+"LBB"
3050 DSKI$ DN,17,SC,A$,B$
3060 IFA$=STRING$(128,255) THEN
3120
3070 PS=1
3080 IFMID$(A$,PS,11)=F1$ OR MID$
(B$,PS,11)=F1$ THEN OD=1: GOTO
3120
3090 PS=PS+32:IF NOT(PS>128) THEN
3080
3100 SC=SC+1: IF SC>11 THEN 3120
3110 GOTO 3050
3120 RETURN
3130 FORSC=3TO11
3140 DSKI$ DN,17,SC,A$,B$
3150 IFA$=STRING$(128,155) THEN
3210
3160 FOR PS=1TO97STEP32
3170 IFMID$(A$,PS+8,3)="LBB" THEN
PRINT MID$(A$,PS,8)
3180 IFMID$(B$,PS+8,3)="LBB"THEN
PRINTMID$(B$,PS,8)
3190 NEXTPS
3200 NEXTSC
3210 RETURN
```

*EDITORS NOTE: This program was
formerly sold by FARNA Systems*

The Hardware Hacker

Dr. Marty Goodman

Not just questions and answers, but complete hardware guidance!

A Floppy Disk Drive Tutorial

Terry Simons, publisher of the disk-based Color Computer newsletter UP-GRADE (Mid Iowa & Country CoCo Club), asked me about some problems he was having writing to diskettes he had bought. Seems his drives appeared to write to and verify the information, but when he sent them to other people, a number of them reported they could not read the diskettes. My first thought was that the drives writing the diskettes were out of allignment. Terry checked his drives with a J&M disk drive allignment checker program ("Memory Minder") and found them to be in generally good allignment. There seemed to be more people having problems than would be expalined by the occasional end user with badly alligned drives.

It turned out that Terry purchased diskettes that were designed for use with a 1.2 megabyte "AT style" 5.25 in drive. That explained the unreliable writes to the diskettes, because such diskettes have differnt magnetic media (with twice the magnetic coercivity) from disks used for 360K drives.

In the course of our discussion, Terry and I had trouble with terminology, because words used to describe the differing diskettes and disk drives tend to be very ambiguous. So here are some clarifying comments I offerred to Terry regarding 5.25 in drives:

Most color computer users employ one or another varient of what is bascially the same sort of disk drive. This is the 35 track single sided, 40 track single sided, or 40 track double sided 5.25 in drive. The single sided drives were sold by Tandy up until their last drive system, the FD502. All Tandy drives were 40 track except for the very first drive system they sold, which could only write 35 tracks. Many CoCo users today have double sided drives. Double sided 5.25 in 40 track drives were the

standard for years in the world of the IBM PC, too. Such drives, sometimes called "double density" drives, could put 360K of data on a diskette after formatting. The 35 and 40 track single and double sided drives all use the same type of diskette. This diskette is a 5.25 in "double density" diskette that has magnetic media with a coercivity of 300 oersteds. The defining character-istics of these drives is that they use magnetic media (diskettes) that have a magnetic coercivity of 300 oersteds, and that they read and write data at a rate of 250K baud.

Some OS9 users and some ADOS us-ers have been using 5.25 in drives that can write 80 tracks, and put a total of 720K of formatted data space on a double sided disk. These 80 track, 720K drives are rather unusual in the greater world of computing. The Tandy 2000 used them, as did a few other ancient, obsolete MS DOS engines, and a num-ber of CPM computers used this type of drive, too. These drives, very con-fusingly called at one time "quad den-sity drives", use the SAME type of dis-kette as those used by the 360K drives. Their data transfer rate is, like the 360K drives, 250K baud. They have the same number of sectors available on a given track. They differ only in that the tracks are half the width, so there can be 80 tracks on a side, instead of 40. To read a 35 or 40 track disk, the head is "double stepped", or moved two tracks at once. The same rule should be observed when writing as between 360K and 1.2 meg disks: only write to a disk formatted in the writing drive. OS-9 provides for double stepping the drives. Disk BA-SIC users would have to add a circuit to allow double stepping (see June 1989 Rainbow, page 98).

When IBM introduced the PC AT, they supplied with with, as its "standard" drive, a 5.25 in drive that had a total data capacity of 1.2 megabytes. This drive is best referred to as a "1.2 meg" drive, or as a "AT style 5.25 inch" drive. This type

of drive was patterned after the last generation of 8 inch disk drives, for both it and its 8 inch predecessors had 80 tracks which each had on them capa-bility to hold 15 512 byte long sectors per track.

The "AT style" drive has a data transfer rate of 500K baud, twice that of the 360K and 720K drives. It uses DIF-FERENT diskettes. These disks are to-day commonly (and very confusingly) called "high density" 5.25 in diskettes. One would better refer to them as "1.2 meg" diskettes, for sake of clarity. These "1.2 meg" disks have a magnetic coercivity of 600 oersteds. This means that it takes MORE magnetism to magentize them than with a 300 oer-sted disk. "AT style" 5.25 inch disks will NOT work reliably in a 35 or 40 track single or double sided drive of the sort used by most CoCo owners.

On PC compatibles, one can format a totally blank, unformatted 1.2 meg dis-kette in a fashion that puts only 360K of data on it, and that allows it to be read by most 5.25 inch 360K drives. Simi-larly, a 1.2 meg disk drive can be used to READ 360K type diskettes. Prob-lems arise when one trys to WRITE to a 360K disk that was formatted on or written to by a 360K drive, using a 1.2 meg drive. The general rule is to format and write the disk with the same type drive and no other.

99% of CoCo systems that use a Radio Shack floppy disk controller or one of the third party clones of same (J&M, Disto, HDS, etc.) cannot use 1.2 meg 5.25 in drives (and cannot use 1.4 meg 3.5 in drives) because such controllers are inacapable of supporting the 500K baud data rate required by such drives. Such ordinary CoCo systems CAN use 360K 40 track 5.25 inch drives, and can also use 720K 80 track 5.25 and 3.5 inch drives.
(Editor: The oldest Tandy 12V con-troller can be modified for 1.2 meg drive support... watch for an article!)

Terry's problem was that he was trying to format 1.2 meg diskettes in a 360K drive. The improper magnetic coercivity of those diskettes resulted in unreliable formating and writing.

Some other comments: 1.2 meg drives normally rotate at a different speed from 360K drives. They usually are set up to rotate at 360 rpm, instead of the 300 rpm standard for 360K drives. The 1.2 meg drives normally rotate at 360 rpm both when they are writing 1.2 meg diskettes AND when they are being used to write a pseudo 360K diskette. However, most brands of 1.2 meg drives have the CAPABILITY to rotate at 300 rpm. One must jumper them into a "dual speed mode" to permit this. Specific information about a given individual model of drive is needed to find the correct jumper settings. There are some files on Delphi that give such information, and one file that specifically details jumper settings for a 1.2 meg TEAC drive. Perhaps I'll give those details in this column if enough requests are made (these files are on this month's "microdisk").

There are many other more obscure disk drive types that I did not discuss here. These include the 80 track 5.25 in 360K single sided drive, the Amdek 3.0 in drives, and a number of 3.25 in drives. Note that Apple and Commodore drives CANNOT be used with a CoCo, for they have totally different logic boards and interfaces.

Disk Drive Summary:

Diskette Capacity			Xfer Rate		Diam	Magnetic Coercivity	CoCo	
Tracks	Sides	K bytes	K baud		inches	Oersteds	Compat	NOTES
35	1	157.5	250		5.25	300	Y	1
40	1	180	250		5.25	300	Y	2
40	2	360	250		5.25	300	Y	3
80	2	720	250		5.25	300	Y	4
80	2	1200	500		5.25	600	N	5
80	2	720	250		3.5	600	Y	6
80	2	1406	500		3.5	700	see note	7

Notes:

1) ancient, original CoCo disk drive supplied by Tandy. Disk Basic was designed to support this obsolete 35 track drive, and was never altered by Tandy. ADOS and ADOS 3, of course, permit full use of 180K, 360K, and 720K disk drives under Radio Shack disk basic.

2) disk drive supplied by Tandy in all CoCo drive systems after the first, gray cased full height drive, with the exception of the last system released, the FD502 system.

3) original "standard" disk drive for ancient "PC Compatibles". Now considered obsolete in world of PC compatibles, and considered the standard for CoCos.

4) this is an odd-ball 5.25 in drive, used in the Tandy 2000 and some CPM computers. It is electrically identical to 720K 3.5 in drives. It has found favor with CoCo OS9 users.

5) this is the "AT style" disk drive, supplied with the original IBM PC AT. It is gradually becoming obsolete in the world of the PC, as 3.5 in 1.4 meg diskettes and drives take over as the current standard in the world of the PC, with 2.8 meg disk drives and 20 meg floptical drives waiting in the wings.

6) this drive has become popular with current CoCo OS9 users, for it is plentiful and cheap and compatible with existing CoCo disk controllers.

7) this is the current "standard" in PC compatible drives. It is commonly (and quite wrongly) referred to as a "1.44 meg" drive. In fact, its data capacity is actually 1.40625 megabytes, and so it is most accurately referred to as a "1.4" megabyte drive. The error was made when slipshod promoters decided to re-define a megabyte as 1000 kilobytes (a "megabyte" is 1024 kilobytes). This drive cannot be used as a 1.4 meg drive with a CoCo disk controller, but CAN happily be made to function as a 720K drive.

Other Items of Interest

I recently succeeded in converting a **Puppo Keyboard Interface** designed to work only with an Eagle keyboard into one that would work with a PC XT keyboard. The conversion primarily consisted of making a copy of the ROM from a Puppo XT keyboard interface and substituting that ROM for the one used in a Puppo Eagle keyboard interface. One might also want to try adding a 470 pfd capacitor between pin 1 of the keyboard conector and ground, and pin 2 of the keyboard connector and ground.

Alain Pilon asked me about getting a Mouse for his CoCo. He wanted to buy a PC compatible mouse, and then re-wire it for the CoCo. I explained to him that mice for the CoCo were electronically unique (they work identically as a joystick), and that IBM PC type mice simply cannot be made to work via the joystick port on a CoCo. He would have to find a used CoCo mouse. Drivers are available that allow use of a serial PC type mouse under OS-9, and at least one software package under DECB (Diecom's "The Rat" graphics program).

Questions for this column may be addressed to the magazine or directly to Dr. Goodman at 1633 Bayo Vista Ave., San Pablo, CA 94806; or Delphi E-mail to MARTYGOODMAN. Internet users may also send E-mail to martygoodman@delphi.com. Please, do not expect a personal reply.

Dr. Marty Goodman, M.D., is an anesthesiologist, long time CoCo advocate, and electronics tinkerer. He has been writing about the CoCo continuously since 1986. We are proud to have him continue that legacy in these pages!

Beginning With OS-9
Time to take the shrink wrap off that box and start learning!

<div style="text-align:right">*Rick Ulland*</div>

Before we plunge head first into our CoCos, allow me to drag out my soapbox for just a minute. Many users of OS-9 will say, in all seriousness, that you MUST install this or that patch or do some hardware modification to your CoCo before 'really' using OS-9 These same people will inform you, in the same serious tones, that most of the utilities on your Tandy disks are absolute garbage and must immediately be replaced. HOGWASH! It is true that after installing the 30 odd patches available for Level 2, it is a much more capable system than stock- in the same fashion that a bored and blueprinted 350 Chevy with twin carbs and a hot cam is more capable than Mom's Caprice Classic. And while the 'six way power seat' commercial utility programs are nice, there is certainly nothing wrong with pulling the lever and scooting forward. We will be getting into all of that here, but for now lets just get the Caprice to the handy mart! I know kzgen will make a bootfile a tad faster and you really have to strap the MPI to run 3 serial ports!!! That's it for the sermon. On to business.

Tandy, in their infinite wisdom, decided to ship OS-9 set up in its least capable configuration. A single 32 column all cap screen and 35 track single sided drives greet the new user with open arms! The joy a he feels when first viewing this screen is hard to imagine. There is a good side to this, however. It has to be changed right away so we can explore the possibilities. Most operating systems are a single program, designed to operate one single computer one single way. DECB runs a CoCo this way. MSDOS runs an IBM or 'clone' that way. If you don't like it, tough. Buy a program that includes it's own substitute for the resident DOS if you can afford it. But OS-9 isn't a monolithic whole- it's a collection of pieces. Technically, there is no single program 'CoCo OS-9'. Tandy (and other makers of OS-9 machines) took the generic 'OS-9 collection', added some CoCo specific modules, and wrote a manual. Viola! CoCo OS-9!

There are some big advantages to this system. First, you can make as big or as small an Operating System as you need. A 128K single floppy machine can't afford to have big chunks of hard drive code sitting uselessly in RAM, while the power user might not be able to live without his serial mouse or other special purpose driver. When a 'hacker' wants to improve some aspect of the system, he only has to deal with one small module, not hun-

dreds of kilobytes of code. And last, who cares what the maker THOUGHT they were selling- make what you want out of it.

The MODULES directory on the BASIC09/ Config disk includes a pretty good selection of OS-9 modules. Each file there has an extension which gives you some clue as to what it is. Tandy also supplied a bootlist.This is just a list of the modules you want in your system. It helps to think of the bootlist as an outline. The stock Tandy bootlist reads:

OS9P2
Init
IOMan
RBF.mn (Random Block File manager, can handle any disk drive style file system)
CC3Disk.dr (drives common coco floppy drive systems)
D0_35s.dd (35 track single sided drive)
D1_35s.dd (35 track single sided drive)
DDd0_35s.dd (default drive, 35 track single sided drive d0)
SCF.mn (Sequential Character File manager does the one character at a time things)
CC3IO.dr (CoCo 'console' driver handles the vdgint.io, keyboard, and video, using these)
VDGInt.io
GrfInt.io (two IO files)
Term_vdg.dt (the 'main' screen- a 32x16 VDG!)
Term_win.dt (40/80 column window screen)
W.dw (Generic window uses next available numbered window. Or call by number.)
W1.dw (Each window HAS to have a number!)
ACIAPak.dr (hardware serial port driver)
T2.dd (describes Tandy RS232 pak/clone)
PRINTER.dr (bit-banger printer driver)
P.dd (bit-banger printer descriptor)
PipeMan.mn (manages communication between process)
Piper.dr (driver for coco hardware)
Pipe.dd (and descriptor)
Clock.60Hz (can't timeslice without a clock!)
CC3Go ('launches' OS9 on coco3)

Using the stock bootlist as an outline, and our notes, we can pick apart the stock list, and make up our own personal wish list. All it takes is a text editor! Lets run through the stock list section by section:

KERNAL: There are 3 modules added automatically to your boot (REL, Boot, OS9P1... won't be in the bootlist) which form the heart of a basic OS-9 system. The two modules at the start of bootlist are extensions of this basic

set and should always be included. These five modules are collectively known as the kernal. This inner core only needs 2 external modules to form a 'real' OS-9 system. IOMan provides the path from here to any of the managers listed below, while clock is needed to time the multitasking. Now we get to the options.

RBF: Almost a no-brainer. Anything that even resembles a disk drive goes through this manager, including floppies, hard drives, and ram disks. Under RBF the most common entry would be the CC3Disk driver and the floppy descriptors it needs to run your floppy drives- which are probably better than the 35 single 'standard'. Replace the stock descriptors with ones that reflect your floppies (an FD502 is really a 40trk, double sided drive; FD501 single 40 track). Hard disk owners will add the hard disk driver and descriptor set right after the floppy stuff. RAMdisks fit under this heading also- so the ramdisk driver and descriptor should be placed in this block of the bootlist. There is a special descriptor for RBF- the DD or default drive is simply a copy of one of the other RBF device descriptors, renamed as /dd. /dd insulates your system from your programs... the program doesn't have to know if you run floppies, a hard disk, ramdisk, or something that hasn't been invented yet... it's special files will be on /dd!

SCF: This is the hardest working manager on your coco! The first thing it is responsible for is the console (keyboard/video). Here you have 3 levels to choose from. VdgInt is a very CoCo specific sort of console driver. It's fast- which is why most Tandy games use it. But it's not terribly capable, so most programs that don't come from Tandy use one of the others. It does have the advantage that it will live peaceably with either one of the other console drivers, but unless you plan to run lots of Tandy games from your stock boot, remove it. The second console driver is GrfInt.io. This is the windowing OS-9 console most programs expect. In theory, it sacrifices some speed for function. In practice, you have to have at least GrfInt to do much. And the last console choice is only available to folks who have the MultiVue package. WindInt adds some extra features to GrfInt- many of the fancy menus and bordered windows in MultiVue are built into WindInt! In fact, many programs that 'require' MultiVue really only require WindInt. The descriptor Term_vdg is the 32x16 CoCo1 text screen. Immediately replace it with Term_win.dt, which can easily

be modified to become an 80 column screen. 128K users may find this is the only 80 column window they can 'afford'. The rest of the descriptors (the w#.dw series) should all be included. They don't take up much room, and you will need a numbered descriptor for each window you (or a program) opens.

Next thing up is the communications ports. If you have an RS-232 pack, the official driver is ACIAPak. There are 2 descriptors - /t2 for the 232 Pak, and /m1 for a modempak or RS-232 modified to use the modempak address. The second choice sort of made sense years ago - SIO and /t1 can do 300 baud through the bitbanger port. Totally useless today! Remove them from bootlist so they can rest in peace. Printer and /p use the bitbanger as a serial printer port. Not much to say about that - if that's where the printer cable ends up, you need this pair!

PIPEMAN: One of the main things OS-9 adds is the ability to have all this independent stuff running at the same time. Which brings up the musical question, what if I have to connect 2 of these programs together? This is the set. Even if you haven't used this group yet, keep it around. To make up a new boot, you first draw up a new outline similar to the one above. Most changes are made from right to left - first looking at the descriptors, then the drivers, then perhaps the managers. If a new descriptor will do, just replace the old one with the new.

When removing modules, first pull out the unneeded descriptors, then any drivers that no longer have anything to drive. So if the 300bps bitbanger driver /t1 goes, its driver SIO can go, since there is nothing left in it's section of the outline. When adding new system software, work from left to right. Figure out which manager is responsible, then add the new driver and descriptor(s) in that manager's section. In general, anything that looks like a disk drive (hard disk, ramdisk, etc.) is RBF. Tapes, terminals, and modems are SCF.

There are certain limitations on how a bootlist can be set up, and these are the main guidelines: The entire OS-9 system has to fit in a single 64K 'process space'. This size is dictated by the address range of the 6809 itself - switching GIME registers while 'inside' the operating system is not allowed! Certain utilities use this 'system ram' for data, most notably format, which uses a whopping 6K! When you get an error 237 (System RAM full) the amount of stuff in your bootlist, plus any data space system utilities are using has went over this limit. The only solution is to

remove some stuff from boot. A few likely suspects were mentioned above (SIO, /t1, vdgint) and there is a third. CC3Go has no reason to be in os9boot. It's only used once, when the CoCo starts up, then it's stuck - being mushed into the boot, there's no way to get rid of it! Put CC3Go in the root (main) directory of your boot disk, and delete it from bootlist. OS-9 will find it, use it, and delete it from RAM when done.

The second consideration is squashing a collection of problems known as the BLOB, or Boot List Order Bug. The symptom is a boot that just won't, even though it looks perfect. There are a lot of theories and fixes for the BLOB, ranging from total voodoo to some well thought out cures. The thing to remember is the BLOB is a whole collection of problems, so there is no 'One True Fix'. Position is critical. The GIME splits a CoCo's RAM into 8K blocks. Managers like to find the drivers and descriptors they manage in the same 8K block of RAM as they occupy. By keeping the outline format above, there is a better chance that each subsection will end up in the same block. Sometimes a section will start too close to the end of a block anyway, and will 'run over' into the next one. If an otherwise good looking boot just will not work, stir! There are 2 modules that don't really belong anyplace - try moving Init or Clock to the other end of the list. Another possibility is changing the order of the three main sections RBF, SCF, and PIPEMAN. One fact to remember is, once another boot is made up, all bets are off! Don't automatically include this or that 'fix' just because it was needed last time - not only may it not be needed, but may cause the problem.. After a few years you end up with a fix to fix the fix that fixed the original problem, which might have disappeared by itself on the new boot!

Now to actually make a boot disk. If you are like me, the first thing you did was read the Getting Started section of the manual, which says use Config to make new boot disks. I did, and frankly Config is a terrible program. The intention is good, but there are just to many limitations. Instead, use that most cryptic pair of OS-9 utilities - OS9gen and Dsave. WAIT!! COME BACK! Neither is as hard to use as the manual leads one to believe. OS9gen only wants one thing - the bootlist we have been talking about. Chd to the modules directory on a copy of your B09 disk. Copy the stock list to another file, using a name that will mean something six months from now! It's a pain to wade through a dozen files with names like boot2.1a, looking for the one that supports double sided floppies, speech pak, and Tandy game gfx. Much better to put up with the

extra typing now and call MbootVDGdsSSP. Using any word processor or text editor (OS-9 has Edit with it), edit this copy to reflect what you want in the new boot. Double check the final version against the description above, just to make sure everything is in there, and in the right place. MultiVue owners should replace GrfInt with WindInt, and add the extra window descriptors w8 through w15 (remember to copy the modules themselves from the MVue modules dir to this one). With the new bootlist made up, put the B09 disk in drive /d1, and a freshly formatted disk in /d0; chd to /d1/modules, and then type os9gen /d0 <newbootlist. This will put exactly one file on the new boot disk - os9boot. To get the rest of the System Master on it, put the Tandy System Master disk in /d1, chd /d1;chx /d1/cmds, then dsave /d1 /d0 ! shell.

Single drive users have a problem here. Dsave can't do single drive backups. There is a way, as long as the new boot is smaller than the original. If you have removed the three modules we spoke of earlier, this will probably work. Start with a backup of your old System Master. Delete OS9boot. Insert the b09 disk, chx /d0/cmds;chd /d0/modules, then os9gen/d0 -s <newbootlist. In either case, if you have removed cc3go from the bootlist, make sure to copy it from the modules directory to the root directory of the new Master. Otherwise your CC3 won't go!

Next time: What Tandy didn't sell you. I'll also answer any questions received about starting OS-9 I can be contacted through the magazine, Delphi (RICKULAND), Internet (rickuland@delphi.com), or good old U.S. Mail (Rick Ulland, 449 S. 90th Street, West Allis, WI 53214). Remember... the only stupid question is the one you didn't ask (I won't print your name if requested...)! If I didn't explain something fully enough, and you can't seem to figure it out, write! That's what this column is all about... teaching you how to get started with OS-9.
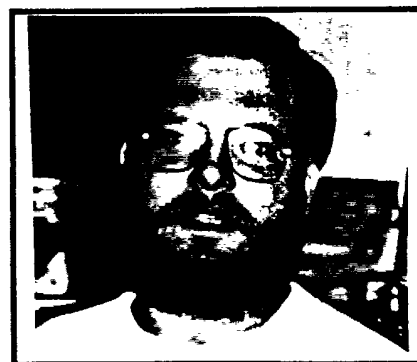
Rick Ulland is the owner of CoNect. He is an avid OS-9 user and hardware builder. He currently uses a CoCo 3 system.

OS-9/OSK Answers!

Joel Matthew Hegberg

Using a mouse with CoCo BASIC09...

Oh my! Is it summer already? It seems to have come so fast! Time sure flies when you're having fun, and with all the recent CoCoFests, that exactly what a lot of us have been having! It's great to see our computers and operating systems getting so much support with so many gatherings.

One of my favorite parts of the fests is seeing what other people have been doing with OS-9 and learning/teaching new tricks to one another. I generally deal with large applications, but applications are just a consolidation of a great deal of software techniques that an author (or programming team) has learned. What I hope to accomplish with this monthly column is the sharing of software techniques and algorithms that are useful under OS-9, regardless of the programming language being used. Be it Assembly, BASIC, C, etc., I welcome any comments or suggestions that you may have for other OS-9 programmers.

Additionally, any questions you may have regarding OS-9 applications programming are welcome. I've written a variety of applications under OS-9, including a graphical word processor, a graphical desktop, graphical clock, games, utilities, and more. I will do my best to answer your questions as simply as possible, although technicality is inevitable with this type of column.

For those readers who are not yet into OS-9: Believe it or not, I know how you feel (Really, I do!). It took me five tries to get into OS-9. The first four times each ended up in failure and frustration, and a vow to give up on OS-9 I remember attending fests when Marty Goodman, during his seminars, would ask all the OS-9'ers to raise their hands, and I would silently scowl at the growing size of the OS-9 user base. I didn't see anything so great about OS-9, yet more and more users were migrating to it, and with few complaints. Let me say if you just don't like OS-9, but have never really been able to play around with it yet, you're missing out. It's that simple. I would have never believed it myself, being the devout DECB programmer I once was, but it really is better to have an operating system that can do a lot at one time, even if there are fewer "flashy" programs available for it. What OS-9 lacks in flashy programs, it makes up a hundred fold in work efficiency.

For you OS-9 readers out there, I know a lot of you are saying, "I love using OS-9, but I can't seem to write my own programs under it." Well, you've turned to the right page, my friend! Write in any problems you may be having, including source code snipits, and we'll set you on the right track! After all, that's what this new magazine is all about — Support! Now, on with this month's topic...

Making your CoCo BASIC09 programs mouseable!

Over the past couple of years, a few people have asked me how to use the mouse under BASIC09. Unfortunately, the GFX2 module does not provide any mouse-reading capabilities. To read the mouse you must use OS-9 system calls, which can be a little confusing to the average BASIC09 programmer. This article may be difficult for a non- or novice-programmer to comprehend, but I believe an average programmer will understand.

To read in mouse data, the I$GetStt (GetStat) system call is used along with the _GS_Mouse call (which is incorrectly labeled _SS_Mouse on page 8-123 of the Technical Reference manual). When this call is used, a 32-byte mouse data packet is sent to your program for examination. Here are some important offsets inside the mouse data packet:

Offset	Size	Data Contained
0	1	pt_valid... 0 if user is in a different window.
8	1	pt_cbsa... 1 if mouse button A is pressed.
9	1	pt_cbsb... 1 if mouse button B is pressed.
24	2	pt_acx... Mouse's X-position.
26	2	pt_acy... Mouse's Y-position.
(values in decimal)		

When using "syscall" (for information, see pg 11-166 in the BASIC09 manual) and the _GS_Mouse windowing function, you must load the X-register with the address of the area you have reserved for the mouse data packet to be loaded into, which must be at least 32 bytes in length. Also, (looking at pg 8-123 in the Technical Reference manual) register A needs the path number, B needs to be $89 (the _GS_Mouse function code), and Y is the "port select". This sounds a lot more complex than it really is. The path number should be 1, which refers to the current window a program is running in, and "port select" should always be 0, for automatic selection, since you do not know if the user has their mouse in the left or right joystick port. To get the address of your reserved area, use the ADDR function (pg 11-6 in the BASIC09 manual). Here's how to set-up

what I've just tried to explain:

```
TYPE registers=cc,a,b,dp:byte; x,y,u:integer
\ (* for "syscall" *)
DIM regs:registers \ (* for "syscall" *)
DIM callcode:byte \ (* for "syscall" *)
DIM mouse_data:string[32] \ (* 32-byte
area where I want my mouse data to go *)
DIM valid,mouse_x,mouse_y, button_A,
button_B:integer \ (* For next example... *)
regs.a=0
regs.b=$89
regs.x=ADDR(mouse_data)
regs.y=0
callcode=$8D \ (* $8D is the I$GetStt code *)
run syscall(callcode,regs)
```

There, not too difficult. After that is run, "mouse_data" will contain the mouse data packet given to you by OS-9 (isn't that nice of OS-9). Now, all you need to do is PEEK at the contents of "mouse_data" and you can extract all the information you need from the packet. Glancing again at pg 8-123 in the Technical Reference manual, you can see that upon exit, register X still contains the data storage area address that we gave it before we ran "syscall". So, to get the mouse's X- and Y-positions, button, and pt_valid information, you could use:

```
valid=PEEK(regs.x)
button_A=PEEK(regs.x+8)
button_B=PEEK(regs.x+9)
mouse_x=PEEK(regs.x+24)*256+PEEK(regs.x+25)
mouse_y=PEEK(regs.x+26)*256+PEEK(regs.x+27)
print "Mouse data; valid:";valid;" cbsa:";
button_A; " cbsb:";button_B;
print " X:";mouse_x;" Y:";mouse_y
```

Again, not so bad. Realize that this program will not display the mouse pointer on screen, only mouse information.

The best way to learn, in my opinion, is through example, so I'm including a program listing that will show how you can also check for keypresses as well as mouse button clicks without pausing program operation. It is also important not to waste CPU time if the user has moved to a different window (which is

when pt_valid is 0). You do this by going to sleep. The sleep technique is also shown in the program listing (Note: All of the page numbers given in the program listing are in the Technical Reference manual.). This program must be run from a graphics screen. To create the proper screen, use the command:

WCREATE /Wx -s=8 0 0 40 24 0 4 2
SHELL i=/Wx&

where x is replaced with the window number you wish to create. Use the CLEAR key to find the newly created window and run the program from there. This will give you a 40-column (320-pixel) graphics screen.

Routine 10000 merely updates the mouse pointer on the screen (and loads in the mouse data packet into "mouse_data"). Routine 15000 (which calls routine 10000 to update the mouse, and get the mouse data packet) simply waits until the mouse button is not being pressed and the user is in the current window. Routine 20000 will update the mouse pointer, check for a mouse click, check for a keypress, and sleep if the user leaves to a different window. Comments are included in the source code to help explain what's going on each step of the way.

PROCEDURE MouseTest

TYPE registers=cc,a,b,dp:byte; x,y,u:integer
dim regs:registers
dim callcode:byte
dim mouse_x,mouse_y:integer
dim mouse_data:string[32]
dim keypress$:string[1]

gosub 5000
10 gosub 15000 \(* Make sure mouse button is not pressed *)
gosub 20000 \(* Wait until mouse button or key is pressed *)
run gfx2("CURXY",0,20)
run gfx2("EREOLINE")
if keypress$="" then
    print "Mouse click at X:";mouse_x;", Y:";mouse_y;
else
    print "Keypress: ";keypress$;
endif
goto 10
5000 REM Turn mouse cursor on.
run gfx2("GCSET",$ca,1)
return
10000 REM Update mouse position on screen.
regs.a=1 \(* Path number 1, which is the current window. *)
regs.b=$89 \(* Code $89 is the SS.Mouse function, pg 8-123 *)

regs.x=addr(mouse_data) \(* Load the mouse information into mouse_data *)
regs.y=0 \(* Automatic port selection *)
callcode=$8D \(* $8D is the I$GetStt callcode, pg 8-54 *)
run syscall(callcode,regs) \(* Go out and get the mouse information *)
mouse_x=peek(regs.x+24)*256+peek(regs.x+25) \(* Get mouse's X-position *)
mouse_y=peek(regs.x+26)*256+peek(regs.x+27) \(* Get mouse's Y-position *)
run gfx2("PUTGC",(mouse_x)/2,mouse_y) \(* Move the mouse to new location *)
REM Note in the line above that the x-coor is divided by 2. This is only if
REM you are on a 320-pixel (40 column) graphics screen. If you are on a
REM 640-pixel (80 column) graphics screen, get rid of the "/2".
REM Unfortunately, CoCo-3 Windows does not scale the positioning of the
REM graphics cursor, so the programmer must take care of that.
return
15000 REM Wait until mouse button is released and user is in current window.
gosub 10000 \(* Update mouse position and load in mouse data into mouse_data *)
if peek(regs.x+8)<>0 then 15000 \(* Mouse button is pressed, so wait *)
if peek(regs.x)=0 then 15000 \(* User is in another window, so wait *)
return
20000 REM Wait until user clicks mouse button or presses a key.
REM Upon exit, if keypress$="" then the mouse button was pressed, and mouse
REM coordinates are in mouse_x and mouse_y.
REM If keypress$<>"" then a key was pressed, and is stored in keypress$.
keypress$="" \(* Clear out keypress$ *)
20010 gosub 10000 \(* Update mouse and load mouse data packet. *)
if peek(regs.x)=0 then \(* The user is in a different window... *)
    callcode=$0A \(* so go to sleep! Don't waste CPU time! $0A is F$Sleep code, pg 8-35 *)
    regs.x=60 \(* Sleep for 60 ticks, which is about 1 second *)
    run syscall(callcode,regs) \(* Go to sleep*)
    goto 20000 \(* Ok, awake again, go back and see if user has returned. *)
endif
if peek(regs.x+8)<>0 then \(* Mouse button was pressed. *)
    return \(* Exit back to main program. *)
endif
regs.a=0 \(* path #0, keyboard input. *)
regs.b=1 \(* Code 1 is SS.RDY function to see if keyboard data is waiting, pg 8-113 *)

callcode=$8D \(* $8D is I$GetStt code, pg 8-54 *)
run syscall(callcode,regs) \(* Go and check if there is any keyboard data waiting. *)
if regs.b<>$F6 then \(* There is data waiting! *)
    callcode=$89 \(* $89 is I$Read code, to read data from a device. *)
    regs.a=0 \(* Path #0, keyboard input. *)
    regs.y=1 \(* Read only 1 byte/keypress. *)
    regs.x=addr(keypress$) \(* Load the data into keypress$ *)
    run syscall(callcode,regs) \(* Go and read the keyboard data *)
    return \(* Exit back to main program. *)
endif
goto 20010 \(* No keypress or mouse click, so go back and try again. *)

Any questions or comments for OS-9/OSK Answers! may be sent to Joel Matthew Hegberg in care of 68Micros magazine or directly to him at 932 N. 12th Street, Dekalb, IL 60115. You may also reach Joel electronically at the following e-mail addresses:

JOELHEGBERG on Delphi.
J.HEGBERG on GEnie.
joelhegberg@delphi.com on the internet.

Joel Matthew Hegberg has been programming on a CoCo, both in DECB and OS-9, for many years. He also programs for the MM/1. Sub-Etha Software carries many of Joel's programs. Here he is shown trying to answer a difficult question at the 1993 Chicago CoCoFest...

Programming in "C"... an introduction to the language   F. G. Swygert

An editors work is never done! Bill Sgambati, author of "CoCo-C", called just last week and told me he wouldn't be able to start his column with the first issue due to time constraints. Rather than skip the column, I decided to write up a quick introduction to C and why it has become so popular. I will also point out the differences between Bill's CoCo-C and the Microware OS-9 C compiler provided by Tandy.

First, a little history of C. Dennis Ritchie originally designed and implemented C on a Digital Equipment Corp. PDP-11 computer for Bell Laboratories under the UNIX operating system. I don't know if there was ever an "A" programming language, but there was definitely a "B". C has some of the characteristics of B, but is not a direct descendant of it. C was written to be easy to understand and well structured, something that would produce source code that could be easily understood by the programmer compared to other popular languages.

One reason C has become so popular is that it does not deal directly with the hardware it is running on. All I/O must be supplied by functions written for a particular machine. This means that the C source will compile with minor, or in some cases no, changes on nearly any platform. Need to do disk I/O? Call the disk I/O function. The compiler written for you particular computer will then add that function to your compiled code. Only the compiler itself is machine dependant. So all one has to do is become familiar with a particular compiler, not learn a complete new language, to move from one machine to another. Then one can start learning how to implement that machines special features through the compiler's unique functions.

Another reason for C's popularity is that it is not restricted to any one program type- it seems to do almost anything well! It's modularity, generality, portability, efficiency, and few restric-

tions make it an ideal language to learn. In many cases, C's efficiency makes it unnecessary to write assembly language sub-routines. An example is the UNIX and OS-9000 operating systems. UNIX consists of around 13,000 lines of source code. Of those, only about 800 are in assembler.

It is suggested that one pick up a copy of "The C Programming Language" by Brian Kernighan and Dennis Ritchie. Ritchie wrote the first C language, Kernighan helped him refine and add to it. The current edition covers ANSI C and is available at most B. Dalton book stores (published by Prentic-Hall). If the store doesn't have it in stock, they can order it for you. There are some minor differences between the original Kernighan and Ritchie (K&R) standard as used by Microware (Tandy) and CoCo-C, but most people have no problem using the book as a guide. The Microware or Infinitum manuals should clear up any differences when encountered.

The program Bill intends to use for the following articles in this series is a file comparison utility. This is something DECB lacks, but there are OS-9 utilities that provide this function. The excercise being accomplished by the series is an introduction to C programming through writing an implementing a short utility. The source code for this utility should work with any C compiler on any platform. It was developed using the CoCo-C compiler, but other platforms and compilers should present no problems. Whenever possible, any minor changes will be noted before hand in the text. If a few slip through, I'm sure readers will point them out (and hopefully some knowledgable readers will help out with the corrections!).

If you don't already have an OS-9/6809 (Tandy) C compiler, you might be out of luck. It is possible to order one from Tandy Software Assembly through your local Radio Shack dealer. It will be at

the last published price ($79.95... I THINK! may be as high as $99.95... inquire!) and the manual will be a photo copy produced from Tandy's micro-film archives. The disk will be from Tandy's archives also, so it won't have the pretty OS-9 label on it (big deal!). Used copies are going anywhere from $40 to $70 - if you can find one (try CoNect... they occasionally have a copy). Your only other choice is Infinitum Technology's "CoCo-C", which is written for Disk BASIC (DECB). Of course, if you are working with one of the OSK (OS-9/68000) machines, you probably got a C compiler with it. There is a "freeware" version of C for OS-9 68000 also. More details on how to obtain this will be passed along when the information becomes available. I'll also write a mini review of it.

A Review of Infinitum Technology's CoCo-C

The introduction in the manual is hard to beat for a quick run-down of just what CoCo-C is. I quote:
"CoCo-C is a complete programming environment designed to be used on a CoCo 1, 2, or 3 with a minimum of 64K of memory and at least one disk drive. The programs contained within CoCo-C are: an Editor, a C compiler, an Assembler, and a Library Linker. All these programs run under the CoCo-c's Command Coordinator" (really just a menuing system)"for speed and simplicity. The compiler is capable of producing position independant (re-locatable) code, or absolute (ROM-able) code. All programs created with CoCo-C are in machine language and do not require a 'run-time' program for execution. LOADM and EXEC is all that is needed to run a user-created CoCo-C program. The CoCo-C Function Library supports many of the standard C library functions along with several 'special' functions which are unique to the CoCo. CoCo-C also supports mixed programming, so that you can combine C, Assembler, and BASIC commands into one program." End of quote.

C Programming...

Over 90 functions are in the CoCo-C function library. Some functions, however, had to be left out. CoCo-C is far from a small-C though. It has 85-90% of the functions in the Microware/Tandy OS-9 compiler. Such things as floating point math simply couldn't be done while maintaining compatibility with Disk BASIC. There are no undocumented ROM calls used, and CoCo-C or any programs written with it are 100% DECB compatible. It will even run on a hard disk or from a RAM disk, provided the DOS operating the hard disk is 100% DECB compatible as well (it was tested under RGBDOS and with several 35 track RAM disks). One of the stipulations for guaranteed compatibility is that the hard/RAM drive DOS exactly emulates 35 track, single sided, 18 sctor/track, 256 byte/sector "standard" Tandy drives. The program should also work with ROMs that have been modified for 40 or 80 tracks, since the compiler calls these routines from the ROM. 40/80 track operation is NOT guaranteed by the author though.

I haven't tried writing any programs in C using CoCo-C, nor have I tried compiling any of the example programs as of yet. I'm waiting for this series to begin in earnest! What I can say is that the manual is professionally typeset and printed with a laser printer. It looks very good and is extremely well written and easy to follow. Bill definately did his homework on this one! This has to be one of the BEST, easiest to follow software manual I have ever seen! Using the examples given, one should immediately be able to start using the compiler. With just a little research, source code written for another compiler should be easily converted to CoCo-C, as long as one stays within the constraints of the CoCo itself (i.e.- CGA grapohics screens should be convertable, forget EGA or VGA!).

Beginner's Showcase. Continued from page 21. Listing of DATEDTR.

```
5 WIDTH 40:CLS 4:'DATEDTR
6 REM COPYRIGHT P.B. BLACKWELL
& FARNA SYSTEMS, 1993
8 REM FROM AUG 1993 68'micros
10 PRINT"********* Pete Black
well ************"
15 PRINT"****** 1408 1/2
McFadden St. ********"
20 PRINT"****** Paris TN
38242-3210 ********"
25 PRINT"********* JUNE 29,
1 9 9 3
************":PRINT:PRINT
30 PRINT"********* <press a
key> ************":PRINT
35 EXEC 44539
40 CLS 8
45 PRINT"!!!!!!!!!!! CAUTION
!!!!!!!!!!!!!!!!!!":PRINT
50 PRINT" If your system
will not accept the high
speed poke and/or your drives
will not operate at 6ms; EDIT
Line 100 for satisfactory
operation."
55 PRINT:PRINT"----------
------------------
":PRINT:PRINT
60 PRINT" This program
should work, with no
problem, on the CoCo 1 and
CoCo 2. You will, however,
have to EDIT Lines 5 &
100; DELETE Line 110
and EDIT the prompt and
instruction Lines to prop-
erly";
65 PRINT" fit the 32 column
screen.":PRINT
70 PRINT:PRINT"**********
<press a key> *************"
75 EXEC 44539
100 POKE 65497,0:POKE 55232,0:
POKE 55318,20
110 ON BRK GOTO 430:ON ERR GOTO
430
120 L$="L3T5O3V30GF": M$=
"L3T5O1V30B-E"
130 PLAY L$+M$+M$+L$:CLS 3
140 PRINT"...<DATEDTR>..
Change/Add/Delete data":PRINT
150 PLAY L$
160 INPUT"Name of file (do not
use /, ., or :)...";F$
170 PLAY M$
180 INPUT"EXT......";E$
190 PLAY L$
200 INPUT"Drive # .. (0-
5)";D$
210 D=VAL(D$):K$=F$+"/
```

```
"+E$+":"+D$
220 OPEN"I",#1,K$
230 OPEN"O",#2,"NEW/
DAT"+":"+D$
240 IF EOF(1)=-1 THEN
PRINT:PRINT"**** No more data
on disk ****":PRINT:GOTO 280
250 INPUT#1,A$
260 PLAY"T60L1C":CLS 3:PRINT
K$:PRINT
270 PRINT"..... Data
Item....";A$
280 PRINT:PRINT"Press <N>=no
change:: <A>=add data::
<C>=change data::<D>=delete
data <Q>=quit::"
290 I$=INKEY$:IF I$="N"OR
I$="n"THEN 400 ELSEIF
I$="A"OR I$="a"THEN 370
ELSEIF I$="C"OR I$="c"THEN
330 ELSEIF I$="D"OR
I$="d"THEN 300 ELSEIF
I$="Q"OR I$="q"THEN 430
ELSE290
300 CLS 8:PRINT" Old data
name: ";A$
310 PRINT:PRINT"!!!! Are You
Sure? <Y/N>";:PRINT
320 S$=INKEY$:IF S$=""THEN
320 ELSEIF S$="Y"OR
S$="y"THEN 240 ELSEIF
S$="N"OR S$="n"THEN CLS 3
: GOTO 260
330 CLS 6:PRINT"Old data name:
";A$
340 PRINT:PRINT"... Change
data to: ";
350 INPUT X$
360 GOTO 410
370 CLS 8:PRINT:PRINT"New data
to add:";
380 INPUT X$
390 GOTO 410
400 X$=A$
410 WRITE#2,X$
420 GOTO 240
430 CLOSE#1:CLOSE#2
440 KILL K$
450 RENAME"NEW/DAT"+":"+D$
TO K$
460 PLAY M$+L$+L$+M$ : CLS 5 :
DIR
```

Repackaging the Color Computer... *F.G. Swygert*

How many times have you wondered if there were an easy, practical way to rid your desk of the cable mess present with most full blown CoCo systems? Most of the articles the author has read describe mounting the CoCo and a Multi-Pak Interface (MPI) in a full size AT style case. This works well, but is expensive and requires a myriad of cables between the CoCo motherboard and the back of the case.

There are several easier solutions! The main focus in this series of articles will be the easiest route for repackaging. Some light soldering will be required, but not a whole lot. No chips will have to be desoldered, just a few connections to the CoCo motherboard... and even those can be eliminated!

The first order of business when repackaging is to consider what is needed in the system. Do you have a need (or perceive a future need) for a MPI? Can you get by without it? The MPI takes up a LOT of room in any case. There are alternatives to it that are much easier to repackage.

The most obvious is a simple "Y" cable. As long as the cable is kept short (preferably under six inches) there are few problems. Since a PC type or similar power supply is used in most repackaging projects (all presented here!), +/-12V can easily be added to a Y cable for items such as the Burke & Burke CoCoXT hard drive interface. The disadvantage is that only two devices are reliably supported. If running OS-9 with a B&B hard drive, floppy controller, and RS-232 Pak...

Rick Ulland, the owner of CoNect, has been very busy working an expansion device for the CoCo: the "XPander". This board plugs into the side of the CoCo and stands perpendicular to it (like a "T"). The internal slots mount devices over the motherboard. The XPander was designed to fit in a modified CoCo case bottom (which is included). It has two internal and one external cartridge connectors (the external is in the same position as the ROM port!). Everything is arranged in such a way that all are inside the CoCo case. A true RS-232 port is built into the XPander board, making an add-on port unnecessary for most users. Used without the new case bottom, the Xpander is perfect for mounting a CoCo in some PC cases. The entire CoCo/Xpander package with add-on cards (no card in the ROM pak slot) requires only 0.2 cubic feet of space (12"x7"x3"), so it will easily fit in an XT or slim-line case. Check the dimensions inside extra small cases of those odd computers out there!

The Howard Medical Slot Pak (SP) presents some of the same problems the MPI does. The most difficult to overcome is the height of the cartridges. A tall case is required to clear long cartridges such as older disk drive controllers, Tandy RS-232 Paks, and B&B hard drive adapters. A short ribbon cable is necessary to turn the SP (or MPI) on its side 90 degrees to the CoCo.

The DISTO Super Controllers are an ideal solution. They have an internal mini expansion bus and several combination cards, such as the 4-in-1 board (parallel port, serial port, SCSI/SASI hard drive controller, and real time clock). All this is contained in the space of a long type disk controller case. Tony DiStefano still has some of these products available, but there may be no more once the supply is exhausted.

Careful planning will help dictate what type of case can be used. Availability is another. Most major cities now have computer salvage or used computer equipment outlets, such as Marietta Computer Salvage in Georgia (near Atlanta, 404-977-9377). There are also mail order firms such as HK Computer (3959 Beltline, Addison, TX 75244, phone 214-484-6447) and TimeLine (23605 Telo Ave., Torrance, CA 90505, phone 800-872-8878) are also a good source for used and surplus cases, monitors, and disk drives. Prices vary from cheap to high... call or write for price lists. Much of the equipment considered obsolete is perfectly suited for the CoCo (5.25" disk drives, composite monitors, etc.) and in high supply and low demand, creating low prices. Some of the other obsolete items are in demand or rare in the used market and command higher than expected prices.

Pawn shops aren't usually a good source for used equipment. They usually demand high prices, though there are exceptions. You can always make a reasonable offer then leave a name and phone number in case the owner reconsiders...

Local flea markets, the Salvation Army, and other thrift stores occasionally turn up something in a usable case. Prices vary widely! Some of the SA directors put high prices on ANY type of computer equipment, others seem to have some idea of what they are looking at. Owners remember most what they originally paid for that old XT eight years ago...

Ham fests and computer shows are the ideal places to look for used equipment and cases. Many dealers bring out there obsolete or hard to get rid of items to the shows and fests. Many ham users have computers to run their equipment or assist in calculations. The CoCo was once real popular with hams, and many often show up at the hamfest swap meets.

Assess your needs, see what is available, then decide on a case type. Do you need portability? Then try to find a KayProII, 4, or 10 case (as shown on the cover). Just want a neat system without an MPI? The Tandy 2000 case is ideal, and even has a "factory" look to it... who remembers that a Tandy 2000 was IBM compatible and not a "super CoCo" anyway? If a PC case is adopted, make sure the power supply is raised for the motherboard to slide under it. If an MPI is necessary, a full size AT case may be required.

The next consideration is the keyboard. Most cases will require an extended keyboard cable. One might be able to eliminate this step if mounting in a Tandy 1000EX/HX or Apple II case, but even a Model II/3/4 case will need a short extension cable.

The connectors aren't as hard to get as one would expect... a little ingenuity must be used! Use a 34 pin card edge connector for the keyboard mylar strip and a 16 pin single row header, half a 40 pin chip socket, or 20 pin single inline chip socket to press in the motherboard female keyboard socket.

Separate the wires on each end of a length of ribbon cable (up to six feet long) for about two inches. Press the wires into a crimp-on card edge connector so that only one side of the connector is used. Cut a 1"x1 5/8" piece of card board thick enough to firmly press the mylar strip of the CoCo keyboard against the contacts of the connector. Note that only one side of the mylar strip makes contact, and that one pin is unused (this one is shorter than the others). The smooth side of the strip makes no electrical contacts.

The other end of the cable is a little trickier to make. If using heasder (row of double ended pins) it isn't hard... just solder the ribbon cable ends to one side of the pins. If using a chip socket, first strip about 1/8" of each wire then tin them with solder. Place a bit of flux in the socket holes or on the ends of each wire. Insert the tinned ends into the socket one by one and carefully touch briefly with the soldering iron. This will be the weak point in the system. If at all possible firmly tape the ribbon cable inside the case to prevent pulling out of the motherboard connector. The preferred method is to make a short cable between the motherboard and a male DB-25 connector mounted on the case, then use a matching female DB-25 on the end of the keyboard cable. One final note- make sure the lines on both ends of the cable(s) all line up correctly! The unused pins on the DB-25 can be used for keyboard mounted power LEDs, reset button, etc.
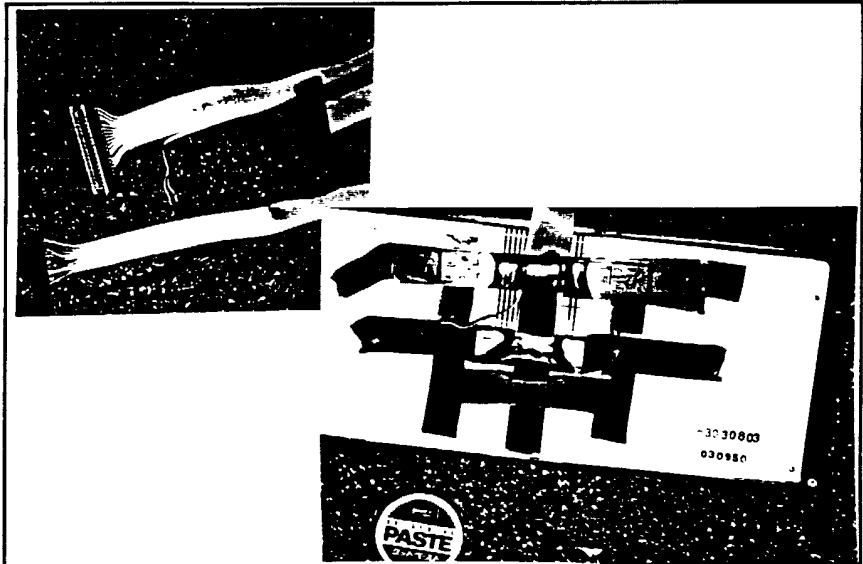
Completed remote keyboard cables. Note how ribbon cable wires are spread apart. Firmly tape the mylar strip and connector to the back of the keyaboard!
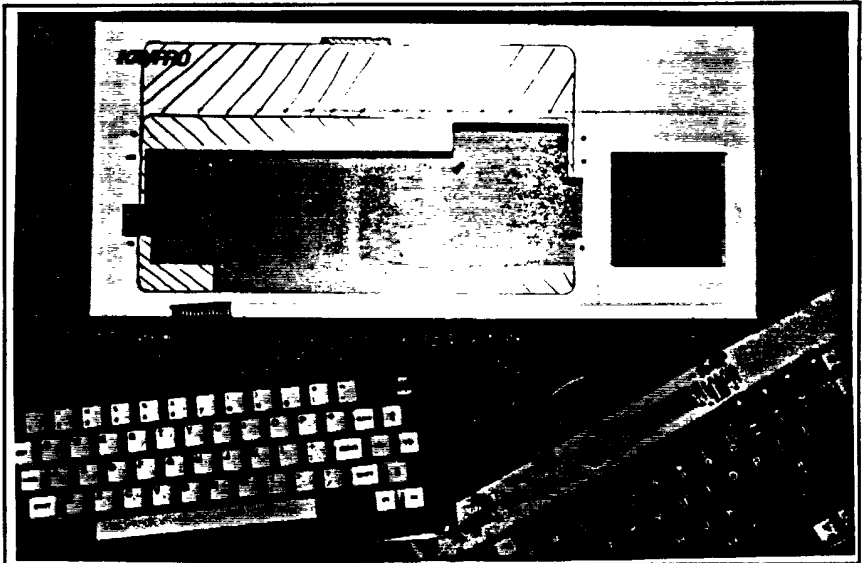
CoCo keyboard going in a KayProII housing. Note two marked cutouts. Upper cutout allows wrist support at bottom of housing, required notching monitor bezel for clearance. KayProII keyboard will give up keys for a future number pad!

Now what about a case for the keyboard? Many variations of wood cases have been used. Some use a modified case from another computer keyboard, such as was done for the portable KayProII/CoCo3 conversion. The original CoCo case can also be used. FARNA Systems has some CoCo 3 keyboards and cases available for $12 (includes shipping) if one desires to keep the original case. I have seen CoCo cases used intact and others with the back hacked off and covered over with cardboard or sheet plastic!

If making a remote keyboard and not repackaging, replace the original keyboard with a wood or cardboard cutout. This is an ideal mounting place for an external connector, front reset button, and power LED.

In the next issue we will go over connecting a PC type power supply to the CoCo motherboard and detail installation in a KayProII portable case. A Tandy 2000 and PC type cases will be discussed in the third installment of this series.

Beginner's Showcase

DATREDER is a short program designed to read DAT and TXT files directly from the disk. It has an option to either print to the screen or to the printer. Most of the necessary instructions are found in the body of the program; however, a few additional precautions/instructions can do no harm. If your system has trouble with the high speed poke [POKE 65497,0], this should be edited out. Also, if your drives will not work at 6ms, these pokes [POKE 55232,0 : POKE 55318,20] should be edited out also. The high speed poke and the disk timing pokes are found in Line 100. The printer options have been set up for the Radio Shack DMP-105. If you have a different printer, you will probably need to change these options. The printer options are found in Line 380 and Line 390.

DATEDTR allows you to read data from a disk file. As you read the data you have the option to (a) retain the old data, as is; (b) change the name (or number) of the data (of course, you lose the old data); (c) add data to the file and; (d) delete data from the file. With DATEDTR you can go to and remain in lower case mode throughout the program, except when setting up "filename/ext" at the beginning. The same precautions/instructions above (DATREDER) applies to both programs, except that DATEDTR has no printer options.

If you press <Q> or <q> or <BREAK> or you experience an error, the programs will (1) preserve your old file; (2) save any additions you have made; (3) CLOSE your files; (4) clear the screen and (5) give you a DIRectory listing at the point that whichever occurred. I am not sure whether this works in case of a drastic ("FATAL") error. I tried to program in as much safety as I knew how.

If anyone does not want to type in the programs, you may send me $1.00 plus two (2) first class stamps (29 cents), and I'll backup you a copy—better yet—subscribe to FARNA Systems microdisk. Mr. Swygert and the CoCo's need all the support they can get!

Happy CoCoing, gentle beings!!!

P.B. Blackwell
1408 McFadden St.
Paris, TN 38242-3210

```
5 WIDTH 40:CLS 4:'DATREDER
6 REM COPYRIGHT P.B.
BLACKWELL & FARNA SYSTEMS,
1993
8 REM FROM AUG 93 68'micros
10   PRINT"***** Pete
Blackwell    ****  ***"
15  PRINT"*** 1408 1/2
McFadden  St.  ***"
20  PRINT"*** Paris    TN
38242-3210   ***"
25  PRINT"****** JUNE 29,
1993  ****  ****":  PRINT:
PRINT
30   PRINT"*********  <press
a  key> *****  ***
***":PRINT
35 EXEC 44539
40 CLS 8
45          PRINT"!!!!!!!!!!!
CAUTION          !!!!!!!!!!!!!!!!!!!!":
PRINT
50 PRINT"        If your
system   will   not   accept
the  high  speed  poke  and/or
your                drives will not
operate  at  6ms;          EDIT
Line   100   for   satisfactory
operation."
55   PRINT:  PRINT"————
——————————————
":  PRINT:  PRINT
60 PRINT"          This program
should   work,   with
no  problem,  on  the  CoCo  1
and                CoCo 2.  You
will,    however,    have
to  EDIT Lines 5  & 100;
DELETE                Line 110 and
EDIT  the  prompt
and  instruction    Lines  to
properly";
65PRINT"          fit the 32
column        screen.":PRINT
70  PRINT:  PRINT"*********
<press  a  key>
*************"
75 EXEC 44539
100 POKE 65497,0:POKE
55232,0:POKE  55318,  20
110 ON BRK GOTO 340:ON ERR
GOTO 340
120  N=0:  M$="L3T5O3V30GF":
L$=  "L5T5O3V30EDG"
130 PLAY L$+M$
140 CLS:
PRINT".....  Disk   File
Reader   .....":      PRINT
150 PLAY M$
160  INPUT"Name of  file  to
be  read  ";X$
170 PLAY M$
180 INPUT"EXT (do not use /
or    .)";Y$
190 PLAY M$
200  INPUT"Drive  #  (do  not
use    :)";Z$
210   Z=VAL(Z$):K$=X$+"/
"+Y$+":"+Z$
220 PLAY L$:PRINT:
PRINT"Send   to   printer
<1>=no::
<2>=yes:?";
230 INPUT P : IF P=2 THEN
380
240     OPEN"I",#1,K$
250 PLAY L$:CLS 3:PRINT
K$:PRINT
260 IF EOF(1)=-1  AND  P=1
THEN 310 ELSE        IF
EOF(1)=-1  AND P=2 THEN 440
270 INPUT #1,A$
280  N=N+1 :  IF P=2 THEN 420
290 PLAY M$:PRINT:PRINT A$
300 GOTO 260
310 PLAY M$
320   PRINT:PRINT"Total   data
items:";N
330 CLOSE #1
340 PLAY L$:PRINT:
PRINT"Read   another?   <Y>::
End   <N>::
DIR    <D>::";
350 I$=INKEY$:IF
I$="Y"THEN 120 ELSE IF
I$="N"THEN 360 ELSE IF
I$="D"  THEN 370 ELSE  350
360 PLAY M$+L$ : CLS 5 : END
370 CLS 8 :PLAY L$+M$ : DIR
Z :GOTO 340
380 POKE 150,41
390 PRINT#-
2,CHR$(27);CHR$(23)
400 PRINT#-
2,TAB(15);CHR$(15);K$;
CHR$(14):PRINT#-2
410 GOTO 240
420   PRINT#-2,TAB(15);A$
430 GOTO 260
440 PRINT#-2:
PRINT#-2,TAB(15)"Total    data
items:";N
450 GOTO 330
```

```
The Listing For
DATEDTR.BAS
is found on page 17.
Sorry for the inconvenience!
```

Now, at the end, you begin to get nervous. Your friends said you didn't have the guts, and now you think that maybe they were right. You look out the window, and the plane does seem terribly high. You begin to sweat. The instructor gives you the thumbs up sign, and you know it's time to go. You stand at the doorway for what seems an eternity. Just when you thought you were going to chicken out, you jump into the air.

A microsecond of sheer terror, followed by exilaration. The wind roars by as you tumble towards the ground. You wait until the last possible moment to open your chute, as if daring the ground come hit you. The chute opens and you float gently to the ground, landing dead center on the target. Later you tell you're friends it was nothing.

Target Jumper is a game for the CoCo 3. The object of the game is to score as many points as possible by making the skydiver land on a target.

When you run Target Jumper, you are prompted on whether you are using an RGB monitor or composite monitor (TV). Enter "R" or "C" respectively. After a brief pause, the title and option screen appears. First, you are asked what size targets you wish to attempt to hit. Press "H" for huge targets, "R" for regular targets, or "T" for tiny targets. Next you are asked whether you want the wind to be a factor. Press "Y" or "N" for yes or no respectively.

The game screen now appears. The target is the light blue box on the bottom edge of the screen. The wind indicator is also located at the bottom of the screen. It points in the direction the wind is blowing. The length of the wind indicator shows the speed of the wind.

Your airplane flies from left to right along the top of the screen. Press the space bar when you want the skydiver to jump. If you do not press the space bar, the skydiver will be pushed out of the plane when it reaches the right side of the screen.

Press the up-arrow key to open the skydiver's chute. Scoring is based on the height of the skydiver when the chute is opened, so be daring and open the chute low. Be careful, however, because after a certain point it is to late to open the chute and the skydiver will hit with a splat. Also, if you open the chute too soon, you'll receive no points even if the skydiver lands on target. After the chute is opened, use the left and right arrow keys to steer the skydiver onto the target. Both of the skydiver's feet must land on the target in order to receive points. There are ten jumps per game.

The maximum number of points per jump is 11, and the maxiumum score per game is 110. However, if you are playing with the no wind option, all points are divided in half. Also, if you fail to open the chute, you are penalized five points, whether there is wind or not.

So go on and do some target jumping. Remember, only chickens open their chute high!

Kenneth Reighard Jr.
441 Ridgeland Drive
Toronto, OH 43964

```
1 'TARGET JUMPER
2 'BY KENNETH REIGHARD, JR.
3 '
10 REM COPYRIGHT K. REIGHARD JR. &
FARNA SYSTEMS 1993
20 REM FROM AUG 93 68'micros
50   CLS:INPUT"MONITOR     (R/
C)";Q$:IF   Q$="R"   THEN   MN=-1
ELSE IF Q$="C" THEN MN=0 ELSE 50
100 POKE 65497,0
105 GOSUB 325
```
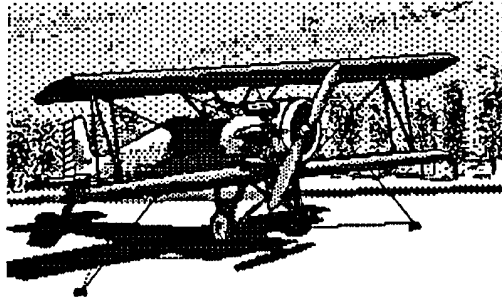


```
110 S=0:JP=10
115 HCLS
120 TL=RND(260)+20
125 IF WF THEN W=.5*(RND(8)-4)
ELSE W=0
135    HCOLOR     7:HLINE(0,186)-
(320,186),PSET
140    HCOLOR     5:HLINE(TL,186)-
(TL+TW,191),PSET,BF
145    HCOLOR     8:HLINE(160,191)-
(160,187),PSET:HLINE(160,189)-
(160+W*10,189),PSET
150 FOR Z=1 TO 1000:NEXT Z
155 FOR X=0 TO 295 STEP 4
160    HPUT(X,0)-(X+25,10),3
165 IF INKEY$=CHR$(32) THEN 175
170 NEXT X
175 HLINE(X,0)-(X+25,10),PRESET,BF
180 Y=0
185 IF INKEY$=CHR$(94) THEN IF
Y<145 THEN220
190   HPUT(X,Y)-(X+22,Y+29),1
195 Y=Y+10:X=X+W/2
200 IF X<0 THEN X=0 ELSE IF X>295
THEN X=295
205 IF Y>192 THEN 210 ELSE 185
210 HCOLOR 4: HPRINT(17,8), "Splat
!!":FOR Z=1 TO 5: PLAY "T255L255
O1ABCD": NEXT Z:SC=-5
215 GOTO 280
220 YY=Y:SC=0
222 IF X<0 THEN X=0 ELSE IF X>295
THEN X=295
225 PLAY"L255T255O1CDCDC"
230 Y=Y+2:X=X+W
240 POKE 343,255:POKE 344,255
245 Q$=INKEY$
250 IF Q$=CHR$(8) THEN X=X-2
255 IF Q$=CHR$(9) THEN X=X+2
260 IF X<0 THEN X=0 ELSE IF X>293
THEN X=293
263  HPUT(X,Y)-(X+22,Y+29),2
264 IF Y>155 THEN HCOLOR 5: HLINE
(TL,186)-(TL+TW,191),PSET,BF
265 IF Y>157 THEN 275
270 GOTO 230
275    IF(INT(X+5))>=TL     AND
(INT(X+17))<= TL+TW+1 THEN PLAY
"L3202T2CD EBFAD":SC=INT(YY/12)
276 IF WF=0 THEN SC= INT(SC/2)
280 S=S+SC
285   HCOLOR   4:   HPRINT(0,0),
"Points=":        HPRINT(7,0),SC:
H P R I N T ( 2 1 , 0 ) , " S c o r e = " :
HPRINT(27,0),S
290 JP=JP-1:IF JP=0 THEN 305
295        HPRINT(13,10),"Jumps
Left=":HPRINT(25,10),JP
300 FOR Z=1 TO 2500:NEXT Z:GOTO 115
305 HCOLOR4: HPRINT (15,
10),   "Game    Over!"
310    HPRINT(11,11),"Play
Again    (Y/N)?"
315    Q$=INKEY$:IF   Q$=
"Y" THEN GOSUB 405:GOTO
110    ELSE    IF    Q$="N"
THEN    POKE65496,0:  CLS:
GOSUB 510:END ELSE 315
320 GOTO 115
325 GOSUB 510:FOR X=0 TO
15:PALETTE X,SK:NEXT X
330 HCOLOR 3,0:HSCREEN2
340   HCIRCLE(25,25),3
345HDRAW"BM25,28D2NL3NR3D3NF5NG5
350   HPAINT(25,25),3,3
355   HBUFF   1,399:HGET(14,11)-(36,
40),1
360 HDRAW"C2BM25,27H7R14G7
365    HCIRCLE(25,19),7,2,1,.5,0
370   HPAINT(25,18),2,2
375   HBUFF   2,359:HGET(14,11)-(36,
40),2
380 HCLS
385 HDRAW"C7BM5,0F5R10F5L20U10
390   HPAINT(7,7),7,7
395 HDRAW"C1BM10,7D2R7E2
400 HBUFF 3,199:HGET(0,0)-(25, 10)
,3
405 HCOLOR4,0:HCLS
410 GOSUB 510
420 HPRINT(12,6)," TARGET JUMPER
! "
425    HPUT(146,80)-(168,109),2
430    HPRINT(8,39),"By    Kenneth
Reighard,    Jr."
435 FOR X=5 TO 1 STEP -1: PLAY "
O=X;L45;BAGFDEC;":NEXT X
440    HPRINT(14,16),"Target    Width"
:   HPRINT(8,17),"(H)uge   (R)egular
(T)iny"
450 Q$=INKEY$:IF Q$="H" THEN TW=
28 ELSE IF Q$="R" THEN TW=20 ELSE
IF Q$="T" THEN TW=12 ELSE 450
460 SOUND 100,1
470 HPRINT(10,19),"Wind (Y)es (N)o
? "
480 Q$=INKEY$:IF Q$="Y" THEN WF=
-1 ELSE IF Q$="N"THEN WF=0 ELSE
480
490 SOUND 100,1
500 RETURN
510 IF MN THEN RGB:SK=11 ELSE
CMP:SK=30
515 PALETTE 0,SK
520 RETURN
```

Rub the Lamp...

As the final issue of Rainbow made it's way into the hands of loyal Color Computer hobbyists, many of them were left asking the now all-too-common question: "What now?" The truth is that many of us have been asking ourselves that question for years, yet here we are. We are still using our beloved "toy" computers and still getting hours of productivity and amusement out of this orphaned system. It seems obvious to me that support will continue as long as we want it. The question is where to get support. There are new magazines coming out, as well as on-line services, such as GEnie and Delphi. The later are the fastest means of support.

Allow me to introduce myself. My name is Allen Huffman, and I have been a loyal Color Computer user/hacker for about 12 years now (yes, my CoCo 1 still runs great, lowercase board and hard-wired 64K upgrade intact!). My support-line was nearly severed in 1984 when I was relocated from Houston, Texas, to a small East Texas town of less than 300 people. I lost all access to bulletin board systems, CoCo clubs, or anyone who even had a computer for that matter. Only the pages of The Rainbow kept me informed.

Through The Rainbow I learned of a "new" on-line service that Falsoft would be working with: Delphi. A bulletin board on a massive scale where CoCo owners nationwide could interact, exchange messages and programs, and stay informed. This, sadly, didn't help me since I had no local access to Delphi. So, after several years "lost in the woods", I relocated once again to a larger town. Unfortunately, there was still no Delphi access, but there were Color Computer users on local bulletin board systems (this was actually before the IBM "death grip" had fully taken place and these systems ran on Ataris or TRS-80s). At one point, half of the systems running in this town were actually on CoCos. But those days soon ended as the PC-Compatible crowd slowly took over. Without local electronic support for my platform I decided to look elsewhere for help.

The Lamp Is Found...

Although there were no major access numbers in my town (no Tymnet, Telenet, etc.) I did find out about a lesser-known service called GEnie operated by General Electric. Surprisingly, it *did* have a local access number. I decided to sign up and have a look and see what it could offer. At this time there was actually a fairly active Tandy/TRS-80 area (known as a "RoundTable") on GEnie and I

was surprised to find a small amount of activity in the CoCo and OS-9 "Categories". I went to work interacting, uploading, and doing my best to provoke support. A time later I found myself taking over a new position - that of "COCO-SYSOP" on GEnie.

GEnie, the General Electric Network for Information Exchange, boasts the largest amount of access numbers of any on-line service. It turns out that the GEIS network (which GEnie operates through) is used by businesses. This is the reason my small town had a number. There are many similar areas nationwide with direct dial GEnie "nodes". GEnie further expanded when they began allowing access through the SprintNet (formerly Telenet) network.

GEnie rates are some of the lowest in the industry. In 1990 GEnie launched one of the first flat rate access fees. In July of 1993 GEnie adopted a new pricing policy which would once again show how strong a contender in the on-line market they are. The monthly fee was raised from $4.95 to $8.95 a month (US - Canada slightly higher). Also, the flat-fee GEnie Basic services are no more. But, the good news is that the hourly connect charges to GEnie have been dramatically lowered! The monthly fee now includes four hours of GEnie time and additional hours (up to 2400 baud) are billed at $3 an hour! Many value services are now covered by this fee, such as the Internet mail gateway (no sign-up fee to access Internet mail) and computer databases. Toll free 1-800 access is available for $6 an hour extra, as is 9600 baud access. In fact, if you use the 1-800 for access, GEnie does not even charge higher rates for 9600 baud! You can dial in toll free at 9600 baud for hourly rates plus $6 extra... and since your first four hours are included, think of GEnie as offering you four 9600 baud hours toll free for $6 each. Later hours add up to $12 each.

Unlike some other on-line services, GEnie doesn't charge you "per message" for private mail. You can send as many messages as you want for $3/hour (and remember, the first four hours each month are free). GEnie's many multi-player games (like the Hitchhiker's Guide-ish Federation II space adventure) are $3/hour. Software downloads and the CB simulator (conferences), all $3/hour. Hundreds of services all at your fingertips for low hourly rates.

For Color Computer and OS-9 support, Delphi still has the most to offer. For those

without access, or those looking for additional support or with no local Delphi access, please check into GEnie. I'm sure you'll find it has something to offer - no matter what your interests are. From research to faxes, shopping to weather maps, the magic of the GEnie awaits your discovery.

Signing Up...

You may wish to call GEnie Client Services (voice) at 11-800-638-9636 to locate an access number in your area, or simply set your terminal parameters to 8/N/1 with Half Duplex (self echo) and call 1-800-638-8369. Once "CONNECTed" type "HHH" (to "wake up" GEnie and let it know what baud rate you are calling at). You will see a "U#=" prompt. Here, type in "XTX99188,TANDY" then have checking account or credit card information handy and follow the instructions. If you are signing up by checking account there will be an additional $2/month fee.

Once your account has been established, drop me a note! From any GEnie menu prompt, type "MAIL" to get to the mail menu, select the "send" option, then address the message to "COCO-SYSOP" and let me know you've made it. I'll be more than happy to help you learn the new system. Feel free to check out the Tandy RoundTable - type keyword "TANDY" at any GEnie prompt or "M635" to move directly to Page 635, the Tandy main menu.

In addition to being CoCo-sysop on GEnie, Allen is the owner of Sub-Etha Software.

Allen Huffman at the Sub-Etha booth, 1993 Chicago CoCoFest. Allen is an ardent "Hitchhiker's Guide to the Universe" fan. Some may recognize the symbolism of the bath robe and towel he typically carries during a 'fest...

Welcome to **On-line Currents**, your guide to on-line sources for the CoCo and OS-9. As your host, I'll guide you to databases on Delphi and the Internet. This month, we have a number of interesting files and announcements on Delphi. Quite a few in fact. I'll concentrate on Delphi this month, leaving the slower posting on our Internet sites for next month. I'll list these offerings under the titles used by the posting user/author in the database, adding expanded descriptions only when needed to clarify what the file is, or which machine it requires.

Oh, a bit about myself. I am a registered nurse, a full time nursing supervisor, and a part-time businessman. I operate BlackHawk Enterprises, a company which supports, among other things, the MM/1, CoCo 3 and other OSK computers. It is my intention to remain as neutral as possible about the relative merits of the competing systems, so feel free to send comments if you think I've slighted anyone.

I bought my first Color Computer in 1981. That CoCo I (F board) ran for quite a while as a dual floppy OS-9 Level 1 system, but was finally replaced with a CoCo 3. My Level 2 OS-9 setup included a B&B 30 Meg hard drive setup, until my multipak succumbed to old age! Presently I get along on my MM/1 system. 3 megabytes of RAM, a 105 meg Hard drive, mouse, ViVa 2400 modem and a Star NX-15 printer round out the complete system.

It may seem that I went OS-9 crazy early and stayed sold. That is true. But rest assured that if there seems to be a slant towards OS-9 here, it exists only because of the relative activity in the two databases that month. My CoCo 3 remains in use, if rarely, and I'll do my best to keep you informed! If you want to compare systems, you may reach me at :

Delphi: NIMITZ
Internet : nimitz@delphi.com

The Kingdom BBS : The Prince, (405) 233-3866, 14.4K Baud, 8-n-1

Now, on to the databases! In the OS-9 On-line Databases, we have 17 new files.

PNW CoCoFest Information: Info about the Pacific Northwest CoCoFest. (June 25-26, Port Orchard, WA)
TED 1.0: Text Editor: An editor for the CoCo 3.
IMASTER 1.40 : An Imaging program for the MM/1. Reads IFF, GIF, and PCX files, and writes IFF files. Allows editing, very nice!
SCF Edition 14: Combines Kevin Darling editor patch with improved support for non-sharable devices!
Multi-Iser CRON: Allows multiple crontab files for multiple users.
Write-Right Availability: Announcement by BlackHawk Enterprises
1993 Chicago 'Fest Report: Joel Hegberg's submission of Allen Huffman's report.
DF: Disk Free Space Command: By TIMKIENTZLE for the CoCo 3
TAGEDIT: Scribe Tag-line Editor: for the CoCo 3, by BIGMILL
Print to Serial Printer: on the MM/1, by TIMKIENTZLE
KIX Q&A Info: Information on FHOGG's latest OSK machines
OS9 FAQ, 7th Ed.: Frequently Asked Questions, thanks, JSHEPLER
RELPATCH: Change OS9BOOT Colors : by MODEL299 for the CoCo 3
Terminal 100 Aquisition: a news release from BlackHawk Enterprise
CCUNZIP 1.02: CoCo UnZip Program : The latest UnZip program for the CoCo 3 by CURTISBOYLE, with RMA source code.
Write-Right Demo for MM/1: Demo version of Joel Hegberg's new word processor for the MM/1.
Neat GIF Demo, JUST VIEW IT!: An excellent demo of the features of GIF89a format uploaded by TELENUT.
DeskJet Printer Graphics Dump: By RICHKOTTKE
KIX\20 Product Announcement: FHOGG announces his newest KIX 68020 OSK machine at a nice price!

Found in the CoCo Databases are 5 new uploads for May, 1993. They are :

DEBBIE01: Picture of a scantily clad woman, uploaded by TRAS
Beatles Albums In IMG #2: uploaded by DEANHOLDER
Source for Fast Disk I/O: asm source from CHETSIMPSON
CF83 FORTH: Forth, for DECB uploaded by MDJOHNSON
COCO BBS List: Thanks to PHILLIPT

That's it until the next issue. See you then!

How To Get In Touch With Delphi

There are several ways to get information on Delphi. You can dial 1-800-695-4005 (voice) to talk to a representative. Alternately, you can dial 1-800-695-4002 with you modem (settings 8 bits, no parity, 1 stop bit). At the PASSWORD prompt, type INFO. Mail may also be sent to Delphi via Internet (INFO@delphi.com). Delphi can be accessed through TymNet and SprintNet. Call those representatives for local access numbers.

On 26 May 1993, it was reported that *Tandy sold their computer manufacturing facilities to rival AST* for an undisclosed sum. Tandy will continue to sell computer products manufactured by others.

One new item that drew a lot of attention in Chicago was a *prototype 68000 processor board for the CoCo from Burke&Burke called the "Rocket"* (see cover photo). Unlike previous 68000 processors for the CoCo, this one fits INSDIE the case, not in the expansion port. The 6809 must be socketed, then the "Rocket" 68000 board plugs into the socket. The prototype carried a 68008 (8 bit buss), but Chris Burke says the production unit will be a 14MHz 68000 16 bit processor. The Rocket carries two SIMM memory sockets capable of supporting either 1MB or 4MB SIMMS for a total of 2 or 8 megabytes of memory. The existing CoCo memory is accessable, but not for applications. It can be used for a RAM disk and is also used for video memory. In essence, this is a single board 68000 computer connected to the CoCo, which works like a terminal. For $195 you get the Rocket processor board AND Professional OS-9/68000, a 6309 emulator, and an OS-9/ 68000 assembler. With 2MB of RAM the price is $295. A very good public domain version of C will be available, but due to license restrictions it can't be bundled with the processor board. A socket for a 68/6309 is also on the Rocket. This was included because the emulator runs at about half the speed of OS-9 Level II on a 6809, though the emulator will run in a window of OSK (OS-9/ 68000). To switch processors, one must reset the entire machine. If you need to run a Level II application, boot with the 68/63 09. If you just need to get some information from an application while in OSK, run the emulator. This looks like a great way to extend the life of your existing hardware while getting into OSK for a small amount. Chris did mention that the boards won't be produced unless enough interest is shown... please let him know if you're interested!

Burke & Burke also introduced another new item- *512 byte sector SCSI device drivers* for the Disto HDISK and 4-in-1 hard drive adapters. This allows use of newer, more common PC type SCSI hard drives. A head parking utility, 256 byte sector drivers, and the EZGen bootfile editor are included in the $29.95 package.

Sub-Etha software demonstrated a very good looking *pull down menuing interface for OS-9/6809 and OS-9/68000*. This is a complete system written in C with pull down menus and mouse support. The 6809 and

68000 systems both work alike, meaning easier porting of applications between the two. All that is needed to add mouse support to programs is a call to a library function, which comes with the developers system. The best thing is that software developers can get a system for FREE! Of course there is a small catch... Sub-Etha requests the rights to sell software that uses this system. Not exclusive rights, and they will pay royalties if they decide to carry the software. Sounds like a reasonable deal!

Frank Hog Labs introduced the low cost Kix20, similar to the Kix30. The Kix20 uses a 25MHz 68020 processor instead of an '030. This board was designed to provide a lower cost OSK platform for those who couldn't afford a Kix30, yet wanted more power and flexibility than an MM/1. A Kix20 motherboard sells for $699.95 compared to a 16MHz Kix30 motherboard at $1499.95. A mini tower case, power supply, 1.44MB floppy drive, and cable kit adds $249.95 to either motherboard (DRAM not included... both support 1, 4, & 16 MB SIMMs). See the FHL ad in this issue for Kix20 motherboard details.

Microware had a booth at the Middle America 'Fest AND the Chicago 'Fest. This is not so surprising itself, since the 'Fests typically have a good OS-9/68000 showing (Microware no longer supports OS-9/6809). Surprising was that they cut the price of OS-9000 (OS-9 for Intel 386 or higher proces-

---

sors) by 60%! During the two days of the 'fests only, one could purchase OS-9000 for only $350! Regular price is near $600. Will they show up in Atlanta also? One can only hope (or write and ask)!

Tony DiStefano had a booth in Chicago. He was selling a lot of left over DISTO products, but also had something new- *a 2MB memory upgrade for the CoCo 3!* Unlike the earlier 1MB upgrade, this one requires soldering only two wires to the motherboard. The upgrade consists of two approximately 2" square circuit boards, onr for the memory and one for access control. The control board plugs into the CPU socket, which means one must either socket the CPU or piggyback a socket over the CPU. Socketing the CPU is the better choice, as there is a socket on the control board for a CPU, and if the CPU ever needs replacing the job is much easier. The memory board holds two "three chip" 1MB SIMMs. The "three chip" type draws less

power than the "nine chip" variety. Using the nine chip SIMM may work, but could cause serious overheating problems and is not recommended. In case you wonder if the new 4MB SIMMs would fit, the CoCo does not have the circuitry to support over 1MB and adding such would be difficult and expensive. The 2MB upgrade sells for $99.95 plus $4.50 S&H ($104.45 total).

HIGH SPEED COCO SERIAL PORT 57,600bps under OS-9! Rick Ulland, owner of CoNect, has created something many OS-9 users have only dreamed of- a serial port that doesn't slow them down! The CCIO port uses a National Semiconductor 16553 which was originally designed for Intel 386 and 486 I/O boards. The chip is so fast that mating it with the comparatively slow CoCo bus required some tricky timing circuits. The 16553 buffers up to 16 bytes compared to only one byte for the more common Rockwell 6551 used in most other CoCo serial ports, whic is the reason the port is so fast. Two nine pin ports and a parallel printer port are mounted in a case the size of the original Tandy RS-232 Pack and sells for $130 plus $4 S&H. The paralell port has all the circuitry for bi-directional operation. Rick hopes to eventually write drivers to support the bi-directional capability. There is a lot of PC hardware that takes advantage of such a paralell port...

A letter appeared on several Internet sites that gives the following E-mail addresses:
PRESIDENT@WHITEHOUSE.GOV
VICE.PRESIDENT@WHITEHOUSE.GOV
These are for real! Personal replys won't be sent, but receipt of your message will be.

Burke & Burke, P.O. Box 733
Maple Valley, WA 98038
Phone 206-432-1814.

CoNect, 449 South 90th Street
West Allis, WI 53214
Phone 414-258-2989

DISTO, 1710 Depatie
St. Laurent, QC H4L 4A8
CANADA

Frank Hogg Laboratory, Inc.
204 Windemere Road, Syracuse, NY 13205
Phone 315-469-7364

Microware, 1866 N.W. 114th Street
Des Moines, IA 50322

Sub-Etha Software, P.O. Box 152442
Lufkin, TX 75915
Phone 409-639-3842/815-748-6638

Reviews...

RASCAN Digiscan Digitizer
FARNA Systems, Box 321,
Warner Robins, GA 31099-0321
912-328-7859
($145.00+$20.00 S&H)

Digiscan is, as the name implies, a video digitizer for the Color Computer 3. This hardware/software combo was designed and currently manufactured in Australia and distributed by FARNA Systems.

To use the Digiscan hardware you need a 512K color computer 3, disk drive, and a video source such as a video camera or VCR. Red, green, and blue photographic color filters for capturing the 4096 color and 3-D images with a video camera (B&W or monochrome cameras CAN make the 4096 or 3-D images) are optional and must be acquired by the purchaser. 3-D glasses are required to view the 3-D images.

The digiscan hardware itself is only about 2" by 5" (much smaller than the original unit!) so it easily fits beside your coco. On top of the unit are two knobs, one to control the white imaging level and one to control the black. Power is supplied to the unit by a 9 volt battery which is installed inside the unit. A small piece of cardboard should be placed between the battery and circuit board to help guard against possible shorts.

Connection to the CoCo is via three DIN plugs; one in each joystick port and one in the cassette port, so a multi-pak interface is NOT required.

The software is provided on a flippy disk with the Digiscan and printer drivers on one side and some sample pictures on the other. Before you start the Digiscan software for the first time, you must tell it what kind of printer you are using. This is done with the included program "Setprint". When you run this program it gives you a choice of four printer drivers: Epson, IBM, Tandy 105, and the Tandy 132. After you make your choice the program exits and saves a default printer file.

After booting the Digiscan software, the title screen soon appears and about 15-20 seconds later the control panel appears. Users in the USA will need to press the 'H' key at this point to set the program to 60hz, as the screen will continue to roll if you don't. The control panel is nicely done in black and white hires graphics and clearly shows all of the options available and the current settings.

There are four modes of image capturing to choose from: 320x200x16 color mode, 640x200x16 black and white, 4096 "Flikpic", and the three dimensional image capture. I did not have a video camera with the color filters so I could not test the 4096 or 3-D images, but the grey scale images turn out very good (with a little practice).

The digitizing process is simple but takes some practice to get good scans. To digitize an image one simply provides a video signal to the Digiscan hardware and press 'D' from the control panel. A thin line starts sweeping from right to left across

the screen with the image being drawn as the line goes by. This is when adjustments can be made with the black and white knobs found on the unit to get the right brightness and contrast for the image. When you are satisfied with the image, press <Break> and it will exit to the control panel where the image can be manipulated with the features of the control panel such as color editing.

The manual is a 25 page booklet which fully explains the features of the Digiscan software, image capturing, the IMG picture format, IMG to CoCo Max 3 conversions. For those who are interested in writing their own drivers, the manual describes the control signals from the Digiscan hardware to the CoCo and shows the actual software routines. The manual is well written and should let anyone start capturing images within a few minutes.

I could only find two faults with this software/hardware package. One is the inability to convert 4096 and 3-D images to the color sequencing used by programs such as The Projector (DECB) or View (OS-9). I did eventually find a way to do this, but it is rather tedious and time consuming. The other fault is that you must set your printer baud rate every time you boot instead of saving the baud rate with the default printer driver file.

This package should satisfy any CoCo user. It is easy to use and provides high quality images. This review can be summed up in two words: buy it!

Quinn Granfor

Color Schematic Designer 3.0
Radical Electronics, Box 1350
Saskatoon, SK S7K 3N9 CANADA
306-664-8724
($35.00 U.S., S&H included)

Where did this program come from? This is version 3.0, and I've not heard of it before! The program shows it's maturity also. It is well written and comes with a complete, easy to read manual. In fact, I was so impressed that I decided to review it myself!

Color Schematic Designer is a schematic drawing program that requires a CoCo 3, hi-res joystick adapter, and 512K. The built in mouse driver has user selectable speed and resolution variables. This allows the mouse to be tailored to the users satisfaction. The cursor moves proportional to the position of the mouse. Several keyboard "hot keys" are used in conjunction with the mouse. Up to 64 electronic symbols can be used in any one schematic. A library of 43 ready to use symbols is included. Most users will find that this set contains all they need. If not, a symbol editor is included for making ones own symbols.

The program is very easy to run. Simply insert the program disk and type RUN"BOOT". The default screen uses a black background with a blue grid and red lettering. Colors may be changed by editing a line in BOOT.BAS to reflect the type monitor used (RGB, NTSC [TV], INVerse monochrome, or MONOchrome). Position the mouse where you want a symbol, press "S" to go into

symbol mode, tab through the symbols with the clear key, then press enter. The symbol can then be rotated in 90 degree increments to the desired position and lines drawn between symbols. Multilayer schematics are supported.

A very useful feature is that the program can be exited and reentered without losing data. Press break to exit. The BASIC command line will appear. Any BASIC statement or even a program can be run. Programs must not use any memory outside the normal 64K BASIC area. Writes to low memory will corrupt Color Schematic Designer. The author claims that any Disk BASIC command is usable, so I exited and formatted a disk, probably the most drastic of the commands. At first, I suspected the schematic in memory to be gone. The manual says to type GOSUB 29000 <enter> then RUN"BOOT"<enter> to return to the program. When I typed GOSUB 29000, I was greeted with a UL (undefined line) error. After running BOOT, however, I was asked if I wanted to continue with the schematic in memory. I answered "yes", and the schematic was still there! DSKINI usually trashes any program in memory as it builds a RAM image of the disk before formatting. One must have the program disk in drive 0 to run BOOT.

Four types of printers are supported: Tandy DMP series, standard IBM graphics compatibles, Epson FX/LX compatibles, and Postscript. The program was designed to use the Epson compatibles, as the output is nearly in a 1:1 aspect ratio (1.1:1) when using hi-res graphics mode on a standard carriage. The wide carriage printer produces a perfect 1:1 ratio. This ratio is length to height, so a 1.1:1 ratio means the item is 1.1" wide for every inch tall. The standard carriage Tandy DMP printers produce a 1.4:1 ratio, the IBM graphics a 1.7:1. The author claims that between 0.8 and 1.2 there will be no noticeable distortion. Quality of output is also rated in the manual. An Epson FX/LX compatible (standard carriage) in hi-res graphics mode rates the best, a Tandy DMP wide carriage the worst. The Postscript driver will send a file immediately to a Postscript compatible laser printer (this was demonstrated at the Chicago CoCoFest) or ink jet or to a disk file. The PS disk file can be transferred to any other computer with the capability of printing a PS file.

This program is almost a must for any hardware designer/tinkerer. Future plans include the release of a Printed Circuit Board designer that will produce ready to etch output and a PCB Autorouter that will route traces between components. These two programs require the Color Schematic Designer to work.

Alan Bayco, owner of Radical Electronics, is using the CoCo and these programs for his own electronics projects. Since he had to write his own software, he was considerate enough to offer it to the general public at below development costs.

F.G. Swygert

Submitting Material to 68' micros

FARNA Systems retains rights to print and distribute any and all contributions. The submitter retains rights to distribute, but not to print in another publication without consent of FARNA Systems..

We accept program submissions in any programming language for DECB and OS-9 (6809 & 68000) of any type (games, utilities, etc.). Articles are accepted covering any aspect of Motorola 68xx and 68xxx processors. This includes microcontroller projects as well as alternate operating systems. If there are enough subscribers interested, we will begin accepting programs for alternate operating systems as well.

Submissions should be sent on disk in ASCII and executable formats. A letter desrcibing the program or article should also be included. Submissions can be made to DSRTFOX on Delphi, or dsrtfox@delphi.com via Internet. Media accepted: 5.25" disk in CoCo OS-9 (35/ 40T, SS/DS), IBM (DD/HD), or DECB (35/40 T). 3.5" in IBM only (DD/HD)

Into the future...

What does the future hold for the Color Computer? That is really up to each of you (us!) who still uses it. Other "orphaned" computers are still alive and well, though there are fewer vendors and new products for them. As long as there are enough supporters, there will be CoCo support, both for Disk BASIC and OS-9, in this magazine.

I believe the precedent set at the Pacific North West CoCoFest may be a ready answer to the trend toward smaller 'fests in general. I believe the CoCo has a stronger following than the other "orphans", so inviting them to our 'fests shouldn't be a big problem. Maybe some of the other 'fest organizers will look into this in the future. It would definitely make a larger gathering, and would lower overall costs.

I had planned a series of articles describing possible updates of the Color Computer. These would have included building an entirely new circuit board around a 6309 and a VGA video chip set. To say that this would be expensive would be an understatement! I do believe a "kit" with parts and circuit board could be assembled for uner $250, but some parts would still be required an original CoCo 3, like the GIME for compatibility and a copy of the ROM.

The idea of building a new circuit board had its merits... until Chris Burke came along with the "Rocket".I've said enough about the Rocket elsewhere in this issue. Suffice it to say now that this product cheaply and effectively stretches the life of the CoCo. I realize it won't help Disk BASIC any, but the majority of those sticking with the CoCo appear to be interested in OS-9 more and more.

If you are a Disk BASIC user only, please don't think that you will be abandoned! That is something I do not intend to do. It has, however, become a primary mission of this magazine to get anyone who is interested help with OS-9.

OS-9 IS NOT as "easy" to learn as MS-DOS to the typical DECB user. MS-DOS commands and syntax are close to DECB, while OS-9 comes from the UNIX world... very different from DECB and MS-DOS!

OS-9, however, is not the only operating system available for the 6809 or 68000. I intend to have some coverage of these alternate systems, and also of some industrial OS-9 applications. They should prove interesting.

Francis G. Swygert

ADVERTISER'S INDEX: