OS9 Boot Modules by David Wordell

I have noticed lately some talk about where one can put the *CC3go* module and what *Init* does. To those interested, I offer the following explanation. First lets look at the Init module found in the original Radio Shack OS9 Level II os9boot file. Just to be sure we are all talking about the same version, at the OS-9 prompt, Enter: **Ident -ms init** This should display " 67 $C0 $0B2322 . Init ", . In Appendix A, page A-3 of the OS9 Technical Reference section of the OS9 Level II manual, you will see the Init Module format. Looking at the following "dump" of the init module, and reading the chart on A-3, you can examine it byte by byte.

```
Addr   0 1  2 3  4 5  6 7  8 9  A B  C D  E F   0 2 4 6 8 A C E
----   ---- ---- ---- ---- ---- ---- ---- ----  ----------------
0000   87CD 002E 0016 C081 CC0F E000 0F27 001A  .M....@.L.`..'..
0010   001F 0022 0027 496E 69F4 4343 3347 EF2F  ...".'InitCC3Go/
0020   44B0 2F54 6572 ED42 6F6F F40B 2322        D0/TermBoot.#"
```

Notice on the first row, address $00 and $01 contain $87,$CD, shown in the chart as being the Sync Bytes. You will see this for all OS9 modules. Address $02 and $03 contain $00, $2E, shown in the chart as Module Size. Look now at the last address in the module. It is $2D, which is one less than the module size. It happens to contain $22 but that is not important right now. The reason that the size seems to be one byte larger is that addresses start at $00 but size is the actual number of bytes. We could not very well have a file that contained $00 bytes. This module contains $2E (hex) or 46 (decimal) bytes. Count them, you'll see.

Address $04 and $05 contain $00,$16, shown as Module Name Offset. If you look now at address $16, you will see it contains $49, the ASCII value for the capital letter I. Also note that this is the first letter in the name of this module, **Init**, and you can see it in the ASCII part of the dump also. Lets stay with the ASCII part for now. Continuing, we see nit. Notice that the t is at address $19. On the hex side we see that address $19 contains $F4, certainly not the ASCII value for the letter t. The ASCII value for the letter t is $74. Since this is the last letter of the name of the module, $80 has been added to $74, making it $F4. This is how OS9 knows that this is the last letter in the name of the module. You can get more information about what is contained in address $06 to $08 in Chapter 3, *Memory Modules,* pages 3 - 7. Lets look at the contents of address $0E and $0F, containing $00, $1A, "Offset to Startup Module Name String". Looking at address $1A we see that it

contains $43, the ASCII value for capital C, the first letter in CC3Go. Note also that the o is $EF. This, again, is the ASCII value of the letter o, $6F, plus $80, equals $EF, signifying the last letter in this Name String. Here is where it is decided where you can put CC3Go. I happen to have mine in the commands directory of my hard drive. My Name String starts in the same place but is says "/dd/CMDS/CC3Go" and the letter o is $EF, signifying the end of this Name String.

Lets look at the contents of address $10 and $11, containing $00, $1F, "Offset to Default Mass Storage Device Name String". Looking at address $1F, we see that the first character in this Name String, /D0 is the /. The last one is 0. Again the 0 is $30 plus $80 equals $B0. Now we come to a mistake in the chart on page A-3.

The Language of Choice

by Randy Kirschenmann

Many times I have heard the C programming language referred to as the programmer's language of choice. Just what is it about this language that would invoke this description? Primarily, I would suppose that C's portability is its most endearing quality. Portability is the term used to describe how easily a program which was written for one platform (hardware/operating system) can be converted to run successfully on another platform. Implementations of the C language can be found on as wide a variety of platforms as any other programming language available. Most of these varieties of C are what is known as ANSI C, which is slightly different from the K&R version we use under OS9 on the CoCo III, but still relatively similar when compared to other programming languages, implemented across these same platforms. Recently I ported *grep* to OS9. *Grep* was written in C for the UNIX operating system and had previously been ported to MSDOS. Except for some minor changes to the function declarations I had very little work to do. I needed to provide two compiler specific functions that were unavailable in the Standard C Library and my work was done in about two hours of machine time. including debugging my own code. Everything else I left untouched.

Portability is not the only plus for this language, however. Versatility is another strong point for C. As Brian Kernighan says it in the preface to the first edition of "The C Programming Language", "C is not a 'very high level' language, nor a 'big' one, and is not specialized to any particular area of application. But its absence of restrictions and its generality make it more convenient and effective for many tasks than supposedly more powerful languages." There are only 28 keywords reserved to the language (the K&R implementation that Microware's compiler under OS9 supports) and all of these are not implemented. Consider Basic09 with over 115 keywords, not even including the function calls to GFX or GFX2. How, you may ask, can C be more versatile than Basic09?

C is versatile enough to write operating systems with. The UNIX operating system is about 80% C and the rest (the lowest level) written in assembler. C is used to write compilers as well as more mundane applications. C's versatility comes from its inclusion of bit wise operators, character array types, integer and floating point numerics, automatic data type conversions, address manipulators and its procedural structure. Functionality is obtained in C by writing functions (subroutines) to perform low-level tasks that can be called by other functions to perform somewhat higher-level tasks, which in turn, can be called by other functions. These low-level functions can be easily placed into libraries, in compiled form, or included in source form from separate files, that then can be called from other C source files. In this way, the building blocks of the C language are put together to provide all of the functionality that is ever needed to perform whatever tasks.

Also, C imposes a certain style of programming, referred to as structured programming, which makes for easy to maintain modules being written. The natural flow of control from one function call to the next provides for easily understood programs, that with the proper choice for function and data naming can lend itself to nearly self-documented applications. Of course, this is as much in the choice of the programmer as in being inherent in the language, and even C can be misused to write obfuscated garbage rather than the clear and concise code that one usually finds in C.

Perhaps C is not every programmer's "language of choice". There are some drawbacks to its use, chiefly in that it places many of the responsibilities of type checking and array boundary checking on the programmer rather than on the compiler. Also, effective use of C's addressing techniques, the pointer data types, with umpteen levels of indirection, can be very confusing to the inexperienced programmer. This can cause the introduction of very hard to debug errors with invalid pointers. However, even considering all this, it is still my "language of choice".

Boot Modules *Continued from page 1.*

Notice that it claims that address $12 and $13 contain the "Offset to Bootstrap Module Name String". Address $12 and $13 contain $00, $22, clearly sending us to the / in the Name String /Term. Notice carefully that the m is $ED, clearly indicating the end of this Name String. The next string, B•••t, cannot be part of this. This is clearly what I would call "Offset to Default Window Module Name String". Yes, some will argue that Term is not a window. Either way, it is the screen you boot up in because *Init* points to it. One final address to look at, $14 and $15, containing $00, $27, clearly, this is the real "Offset to Bootstrap Module Name String", Boot. The last three bytes are the "CRC Value check".

I have included a dump of my *Init* module, showing the path to *CC3go* and the fact that my "Default Mass Storage Device Name String" is /DD. Note also that even though /DD appears in a different location, address $10 and $11 clearly point to it correctly. I hope this helps to clear up some of the confusion.

```
Addr   0 1  2 3  4 5  6 7  8 9  A B  C D  E F    0 2 4 6 8 A C E
----   ----  ----  ----  ----  ----  ----  ----  ----   ----------------
0000   87CD 0037 0016 C081 D50F E000 0C27 001A    .M.7..@.U.`..'..
0010   0028 002B 0030 496E 69F4 2F64 642F 434D    .(.+.0Init/dd/CM
0020   4453 2F43 4333 47EF 2F44 C42F 5465 72ED    DS/CC3Go/DD/Term
0030   426F 6FF4 6413 EF                          Boot/.o
```

-- David Wordell;Fidonet OS-9 Echo --

Disto's "No Halt" Controller by Gene Krenciglowa

Seems a lot of people are having different kinds of problems with their Disto Super Controller II. Moving over to interrupt driven hardware may require more than just putting in the appropriate driver in the bootfile.

As a suggestion, try constructing a simple boot following closely the original Tandy boot order. Include cc3disk.irq and a software clock and make sure no other IRQ devices are iniz'd on startup. Have the SCII hooked directly into the ROM/cartridge port. If the SCII works in halt mode but not in no-halt mode, it may come down to the buffer (static ram) and its ports at address $ff74-ff76.

The following RSDOS BASIC code tests the static ram used by the Disto SCII in buffered I/O mode as in the cc3disk.irq. A 'bad' value is probably conclusive. Lack of a 'bad' value is somewhat inconclusive as the code g•es a lot slower than in the disk drivers. My SCII checked out 'okay'. .

```
10  'CHECK DISTO SCII STATIC RAM, USED BY CC3DISK.IRQ
20  AV=0: MX=255:  'VARY AV TO CHECK.  KEEP AV+MX <= 255
30  'AV=5: MX=128
40  POKE &HFF76,0: 'RESET COUNTER
50  FOR I=0 TO MX
60  POKE &HFF74,I+AV:  'STORE VALUE, COUNTER AUTO INCREMENT
70  NEXT I
80  POKE &HFF76,0: 'RESET COUNTER
90  'POKE &HFF74,255: 'CHECK TO CHECK THE CHECK BELOW
100 POKE &HFF76,0
110 FOR I=0 TO MX
120 CK = PEEK(&HFF74):  'READ BACK AND CHECK AGAINST STORED VALUE
130 IF CK<>(I+AV) THEN PRINT "BAD VALUE: I="; I;"  "; CK; I+AV
140 NEXT I
```
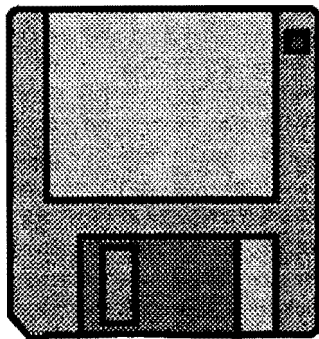
-- Gene Krenciglowa --
House of Fire BBS - Toronto On - (416)601-0085 - (1:250/436)
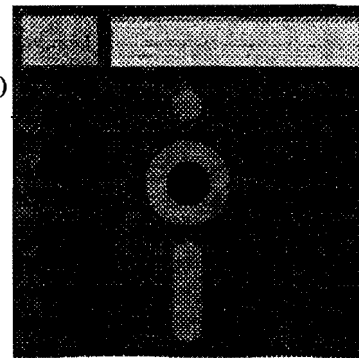
New Products for OS-9

FARNA Systems NEW ITEMS!!!!

OS-9 NVOICE/INVENTORY MANAGER Print your invoices simply by typing in customer information and a part number! Inventory can also be entered in computer listing (does NOT subtract from inventory when an invoice is issued). Prints a customer item list which includes part number, description, and retail price, and an inventory report which prints the above plus your cost! Supports multi-users under either OS-9 Level I or II. Requires Basic-09. Comes complete with source code! License agreement states that you MUST send in any modifications to the source for consideration to incorporate into future versions/upgrades. In return, one free upgrade is returned for any modifications used. Just for participating (even if the upgrade s are not used in future versions) any further upgrades will only be $5.00 each. FARNA retains the right to raise this price upon major version revisions. **$24.95**

RETURN OF THE RASCAN VIDEO DIGITIZER!!!

The Rascan Video Digitizer will convert still video images to computer images. You must have a totally still image, as the digitizing process takes about 15 seconds. Use a camcorder to tape the subject for 30-60 seconds then digitize the recorded image.

Some new VCRs have a digital freeze frame which can be used. Most VCRs freeze are a little fuzzy- fuzziness will digitize also! Features include:
* Plugs into Joystick ports (both)- no MPI needed!
* Brightness & Contrast comtrols *
* Color Burst Signal Filter
* Images can be exported to Max-10, CoCo Max 3, and Color Max 3

* Resolutions of: 640x200 (16 gray levels with dithering), 640x200 (4 actual gray levels), 320x200 (16 colors), 320x200 (4096 colors).

Did I mention it can do COLOR scans? These are more difficult than the normal black & white scans. Color requires taking three shots of the image, each with a separate color filter. The three images are then displayed in sequence 60 times/second to create a color image.

A minor drawback to the system is that images are stretched a little horizontally, making people appear just a little "fatter". Nevertheless, the Rascan was hailed as "the best Color Computer digitizer" by Ed Ellers in the last Rainbow review (December 1990, page 83). Price is $175. The Rascan is actually made in Australia, so allow 8-10 weeks for delivery.

Address order(s) to: **FARNA Systems, 904 2nd Ave., Warner Robins, GA. 31098, (912-328-7859**

* * * * *

LOCATE
by Sirius Software

This is a VERY FAST location utility to find that file that is buried somewhere on your drive!!

For OSK, the 'locate' utility will search a drive containing 100 directories and 1200 files in 16 seconds flat!!, which is considerably faster than the public domain find utility!

For 6809 OS9, the 'locate' utility will search a drive containing 175 directories, and 2250 files in 96 seconds!!, and runs MUCH FASTER than either **wheres** or **ff**.

Sirius Software is offering the locate command at a special price, based on your ability to download this very fast command off of Delphi, Compuserve, or the StG

Network.
For pricing Details, contact JSUTEMEIER on Delphi, 70673,1754 on Compuserve, or jim@zog on the StG Network.

* * * * *

E.N.C.9.

This is a two entry database which consist of a keyword entry and a descriptive entry. Use it to store over 250,000 recipes, names and addresses or definitions, or anything you've seen an encyclopedia used for.

FEATURES:
o IMPORT AND EXPORT FROM/TO TEXT FILES
o PRINT ALL ENTRIES IN DICTIONARY FORMAT
o BUILT-IN LINE EDITOR
o ADD, DELETE, RETRIEVE AND SEARCH ENTRIES
o STORE OVER 250,000 RECORDS USING THE "VSAM" STORAGE FORMAT
o BONUS: COMES WITH A COMPUTER TERMS DICTIONARY

Requires CoCo 3 with 512K minimum, 80 column screen, RUNB, SYSCALL and some standard OS-9 LEVEL II commands. Will be available by January.

CANAWARE *ingenuity from the "great white north"*, 1378 Credit Woodlands Court, Mississauga, Ontario, L5C 3J5, CANADA   L5C 3J5.   (416)279-3037
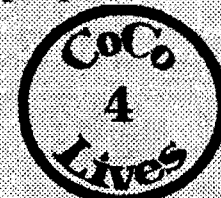
Hardware Hackers Project

Make your own IDE Interface

Below is a file describing the pin outs for an ide hard drive for anyone wanting to try interfacing an IDE hard drive to a COCO

```
pin      description
1        reset
17       hd0
15       hd1
13       hd2
11       hd3
9        hd4
7        hd5
5        hd6
3        hd7
4        hd8
6        hd9
8        hd10
10       hd11
12       hd12
14       hd13
16       hd14
18       hd15
23       HIOW
25       HIOR
27       IOCHRDY
28       HALE
31       IROBUS
32       IOCS16
35       HA0
33       HA1
36       HA2
37       HCS0
38       HCS1
39       SL V ACT
20,29,34      not used
2,19,21,22    ground
24,26,30,40   ground
```

NOTE: You will also have to write a descriptor to properly interface the IDE data with your Hard Drive Driver.

Check the May '92 and August '92 issues of the *OS-9 Newsletter* for the *HD Interface* article on constructing an interface. This should provide you with the necessary information for determining hardware address coding

GOOD LUCK!

Basic09: *using the GFX2 (Select) to change windows*

If the device window **Select** is not done exactly right, it has the nasty habit of locking up the system.. The **Select** must be done on (1) an open path that is (2) the window currently being displayed. A **Close** or a **Dwend** may give the appearance of a **Select** because the system flips to some other window automatically, but this seems to eventually cause other problems, like lockups.

The key **Select** in the Basic09 code below is labelled *key line*. Having that **Select** a couple of lines down after the **Close #win1** does not actually flip the screen (the Close #win1 flips it) and this eventually causes problems if the program is run over and over again without exiting BASIC09. .

This BASIC09 code flips the display window to a /w window and then back to the original display window with redirection to stdout (path 1).

```
.
PROCEDURE dwsel
TYPE registers=cc,a,b,dp:BYTE; x,y,u:INTEGER
DIM regs:registers
DIM win1,Dup,outstd,outdup,wlast:BYTE
outstd=1 \Dup=$82
RUN showscrn(outstd)
regs.a=outstd \RUN syscall(Dup,regs) \outdup=regs.a
PRINT "outdup  "; outdup
CLOSE #outstd \(* free up path 1
OPEN #win1,"/w":UPDATE
RUN gfx2(win1,"DWSET",1,0,0,40,24,0,2,1)
RUN gfx2("SELECT") \RUN delay
RUN showscrn(win1)
RUN gfx2(outdup,"SELECT") \(* *** the key line - before
CLOSE #win1
RUN delay
CLOSE #win1 \(* free up path 1
regs.a=outdup \RUN syscall(Dup,regs) \wlast=regs.a
CLOSE #outdup
RUN showscrn(wlast)
END


PROCEDURE showscrn
PARAM wn:BYTE
DIM zz:STRING[1]
PRINT "path number = "; wn; "  press any key"
GET #wn,zz
END


PROCEDURE delay
DIM i:INTEGER
RUN gfx2("BELL") \ FOR i=1 TO 10000 \NEXT i
END
```

-- Gene Krenciglowa --
Origin: Discus BBS, Hull Quebec, Canada (819)771-3792 (1:163/519

CoNect

CoNect has just come out with their first Catalog of products, Similar to CoCoPro, but with some really fantastic surprises! The following is a list of hardware products being manufactured by CoNect:

CoNect Mouse Tamer Port switch. Exchanges Mouse and Joystick with switchable mouse resolution
Cat#105 $19.95
CoNect RS-232 Port Complete seven line RS-232 port at address $FF68 or $FF6C with normal or swapped Carrier Detect
Cat# 103 $49.95
CoNect Dual RS-232 Dual RS-232 port with FIFo buffering. Available in Jan.
oNect Y-Box Provides physical support of add-ons. Locking plate included (Not a Y-Cable)
Cat# 101 $29.95
CoNect Power-Y Save as CoNect Y-Box but provides +/- voltage for RS-232 Paks
Cat# 102 $39.95
CoNect XPander INTERNAL EXPANSION BUS for the CoCo. 2 internal slots plus maintaining the original ROM Pak slot. RS-232 port included. Includes XPander board, new case bottom for the CoCo and power supply for the inclusive serial port.

The CoCo 4 has arrived!!!!!!
CORRECTION......
Coming Soon!

CoNect and *FARNA Systems* announce "....the first affordable LapTop system Color Computer. For about the cost of a color monitor."
AND THEY CALL IT . . . . .

PoCo

Have you figured it out yet? This CoCo is a laptop! with the following specs:
* 6809 or 6309 CPU
* CoCo-3 Keyboard
* CoCo-3 Motherboard
* 128K or 512K RAM
* 640x200 Display on Liquid Crystal Display
* 2-internal expansion slots
* 1-external expansion slot
* RS-232 (Full support: TD, RD, DCD, DTR, DSR, CTS, RTS)
* 2 hour battery life with AC adapter/charger
     * Optional Dual RS-232 port
     * Optional Internal Trackball (Resolution and port used are switchable)

Following mass storage and modem configurations are possible:
* One 3.5 inch drive and controller mounted internally with external connector for external floppies
* One 3.5 inch hard drive with autoboot with external connector for external floppies
* Dual 1/3 height floppies. (2 floppies or 1 floppy + 1 hard drive)
* Internal Modems, either standard size external modem card wired in and mounted above motherboard or with a hard drive system, only a pocket modem can be accommodated under the keyboard.
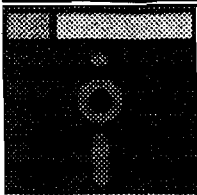
GOTTA GET ONE!

Using 'dEd'

Last month we read how **dEd** was used to restore damaged disk by editing corrupted bits in file allocation table. In this article we see how a perfectly good file is patched and then the CRC is verified.

This "essay" describes how I found and fixed the problem of using a serial mouse via the patches to the **cc3io.io** module which although they work with the serial mouse, also result in floppy disk accesses being locked up till the mouse is moved, including during the bootup if you boot from floppies like I do. These described patches are applicable to the .m51 and .151 versions of cc3io as generated by the script furnished in the file 'smouse.ar' available only on Delphi OS-9 database. I couldn't find the similar patch location for the .m52 and .152 versions and didn't even look at the .joy version. The same logic could be applied to solve those if they are in fact a problem.

In both the .m51 and the .151 versions of the modified cc3io modules, the table used to initialize the interrupt (F$IRQ) will be found starting at offset $B65 as the 'flip byte' of $00, followed by the 'mask byte' of $0F, and a 'priority' byte of $01.

This mask byte value is the problem as it also includes the bit used by the IRQ service flag of the **CC3Disk** module generated for use with the **PCDos** and **RSDos** programs. I suspect this is fixed by the hardware's emulation of a WD-1773 floppy controller chip rather than anyone's idea who wrote the floppy disk drivers and would therefore be applicable to the 'normal' versions of the **CC3Disk.dr**.

I first tried to re-adjust it to $0E since that would free the bit used by **CC3Disk** as reported by 'idir'. Maybe it was a good idea but I still had to play Bobby Bare's record about Madam Rue while slowly rolling the mouse around to get that one booted. VooDoo and all that.

So I checked the status register pinouts in my Deluxe RS-232 manual which said only bit %00001000 was for the receive buffer full flag. It also notes that NO interrupts are generated for the three errors of **frameing, parity** and **overrun** which occupy bits %00000111
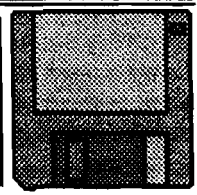
of that status register. So my next fix attempt was by setting the original $0F at $B66 to $08.

I used 'dEd' right on a backup of the boot disk to install the patch. Do this:
> **ded /dn/os9boot**[ENTER]

When the first sectors data is on screen do this:
> **L**

you will be prompted with: 'Link to which module? ' Type:
> **cc3io**[ENTER]

When the first sector of cc3io is displayed (87CD0C7A for .m51), then type:
> **s**

'dEd' will respond with 'SECTOR = $'. Respond with:
> **b**[ENTER]

'dEd' will advance its display to sector $0B of cc3io. Now type:
> **e**

for edit. dEd will turn on a couple of reversed video cursors, one of which is flashing on the first byte of the display.

Using the arrow keys, move the cursor to the byte at $66, which SHOULD BE a $0F, if not, start over by hitting the ENTER key and entering a **u**, then **q** and **y** to get out.

If it is an $0F, enter a **0** and an **8** which will change that byte on the screen and advance the cursor to the next byte. Hit the ENTER key to get out of the edit mode, then a **w** and **y** to write the sector permanently.

Hit the up-arrow to advance to the next sector which will not be a full display in this case. Watch the last 3 bytes, the modules 'crc'. Then enter **v**, dEd will verify all modules in the file, fixing any bad crc's it finds, then reread the currently displayed sector from the disk. You'll see the crc bytes change at that time.

Voila! Its fixed! <Grinning like an idiot>. Now you can boot without unplugging the mouse. Stick that

disk in **/d0** and hit the reset. You are all done. You did remember the **v** above didn't you?

You could also do it with **modpatch** I suppose:

```
l cc3io
c b66 0F 08
v
```

But then you would have to cobbler a new boot. For me, using 'dEd' was MUCH easier. The mouse? Its a Logitek 'DexXa', sells for $20 at WalMart.
<div align="right">Anonymous;Delphi OS-9 Sig</div>

IRQ "HACK" REVISITED

In the November issue of the *OS-9 Newsletter* we featured an article by Eddy Cardone showing how to construct a neat and clean hardware hack to the CoCo rather than to the Deluxe RS-232 PAK. This eliminated the unsightly wire running between the PAK and the CoCo.

I also remember reading an announcement from Tandy 2 or 3 years ago saying that it was not necessary to physically modify the Color Computer, Multipak or the RS-232 PAK. That the IRQ problem only required a simple patch to one of the system modules. PERIOD! That was it! It was like waiting for the other shoe to drop...... What module? What patch? Will it be made available? Good 'ol Radio Shack.

Fortunately we have people in the CoCo Community like Bruce Isted who wrote **SACIA** and **DACIA** to replace the poorly written **ACIA** module, and Eddie Kuns who found the offending IRQ module.

The **clock** module that comes with the system disk (**clock.60hz, clock.50hz**)

GIMIX

80 POUNDS OF OS-9 POWER!!
or a ship anchor?

There have been some inquiries about the GIMIX system that I have for my usage that was purchased by the school district that I work for. The exact description of the two computers that we have is **OS-9 GMX III SYSTEMS.** The computers use our old faithful 6809 CPU and come with five 3 port intelligent serial I/O boards and cables. Each computer has 720K of static RAM, one 80 track floppy drive, one 19Meg Hard Drive and one 6 Meg Bernoulli Box. System software included OS-9 Editor, Assembler, Debugger, BASIC-09, and RUNB. In 1983 each computer cost us about $12,000 (that's my approximation from the 1983 catalogs included with the package)

Beyond the above "basics" there were also 32 Freedom-110 terminals, two dot matrix printers (9-pin), and two hayes compatible external modems. Unfortunately, the modems have "disappeared" and 24 of the terminals were surplused (thrown out!).

Nowhere did I see a specific reference to **OS-9 Level III.** But I did run across the heading **OS-9 II GMX III.** The CPU board does have some customized memory management chips and apparently the machine can access all of the memory contiguously although the 6809 is still limited to swapping 64K blocks or pages of memory. The most noticeable difference is the speed. The clock rate is 2.2MHz which is slightly faster than the CoCo and the 6809 wears a heat sink to handle the increased speed. Actual operation appears to be much faster then the CoCo, especially when you consider I have 7 terminals running at the same time with no apparent slow down of any processing.

One obvious drawback however is the size and weight of the computers (not the terminals). Each of the two GIMIX machines weighs in at 80 lbs. and the overall dimensions are 31 x 34 x 9 inches. The power transformer in the rear of the case measures 8 x 10 x 10 and I'm sure in must weight over 40 lbs.

A ouple of ingenious things are two small circuit boards located in the back of the machine. The first board is a baud rate generator with jumpers to configure the baud rates of the different I/O ports. The second board is a CRC board that computes and verifies the CRC of each module loaded into memory, relieving the CPU of that chore. Boy does that contribute to speeding things up!

Besides the system software that came with the two machines, I found masters for the following software that was purchased separately for the computers:
* Stylograph Word Processor
* DO (script file run time module)
* Print Spoolers by Microware
* Dynacalc with a configuration file used to generate a **dynacalc.trm** to match your terminals
* Sculptor Database
* RMS Database
* DynaSpell (Spell check dictionary, 128K)
* Introl C Compiler and Libraries
* Microware C Compiler and Libraries (same as supplied by Tandy)

I also came across some unusual utilities in the system CMDS directory that I had never seen before, but keep in mind that these systems were designed specifically for multi-user operation and for the most part, only modules in memory are accessible by the average user unless a file or module is created or owned by the user. For example: A user can enter **DIR** and get a listing to the screen at whatever directory he want to look at. This was possible, not because **DIR** was in the CMDS directory, but be ause **DIR** was loaded into memory by the Super User as part of the STARTUP file. The same would be true for most of the standard utilities needed to get around on OS-9. **FORMAT** is in the CMDS directory too but obviously not loaded into memory and impossible for the general user to access.
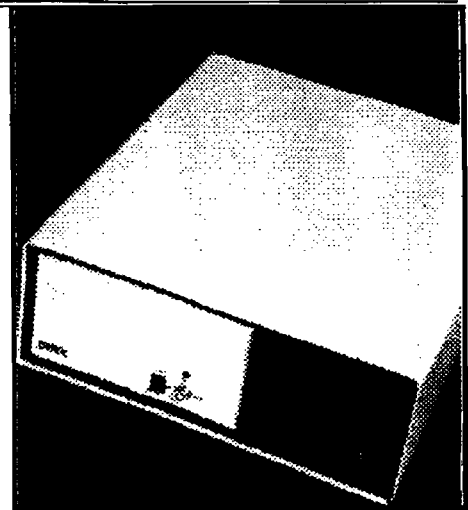
What about the advance user who has access to DEBUG? Fortunately, even command modules in memory can be "write protected" and a utilities in the CMDS directory called **wprot** allows the super user to protect and unprotect individual modules from being modified while they are in memory. Pretty sneaky, eh? There is another utility called **inizd** which I haven't figured out yet.

How compatible is the GIMIX with the CoCo? Well besides the disk format being entirely alien (Standard vs. CoCo) most CoCo utilities work great on the GIMIX. I even loaded **OS9p3** to call up error messages form the SYS directory. No problem. As long as a program or utility does not make a specific call to the "CoCo" screen, everything works fine. However that does mean no graphic programs, no text editors like *VED* and not even *Shell+*. Oh well, I can't have it all, but it sure comes close!
-- Rodger Alexander --
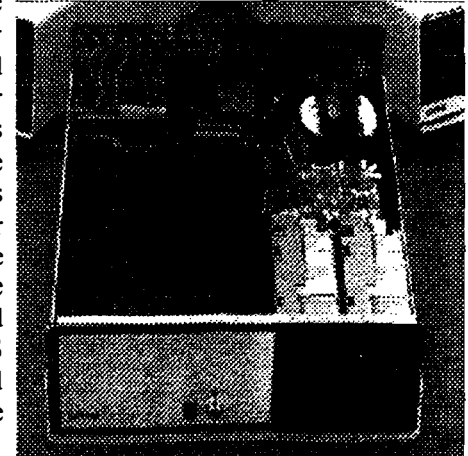
– IRQ Hack – *Cont'd from pg. 7*

needs to provide a constant IRQ pulse to prevent the CoCo from locking up when using the RS-232 PAK. Turns out the same problem exist with the real time clock drivers for the Eliminator, Disto's 2-in-1, 4-in-1 and Burke & Burke's CoCoXT

Solution: Download Eddie Kuns' **CLOCK_UPDATE** archive file on Delphi, Compuserv or your favorite OS-9 BBS (See BBS list on last page for Washington State). If you do not use your computer for telecommunications then you will not need these new modules. The Bellingham OS-9 Users Group's Public Domain Library has both Bruce Isted's **SACIA** replacement module and Eddie Kuns **Clock_Update.AR**. If you would like to obtain the modules from our Library, simply send $1 (75 cents for postage, 25 cents for disk).

-- Rodger Alexander, Editor --

Bellingham OS-9 Users Group
Public Domain Library

The following files have been added to the PD Library during the month of December 1992.

Multivue Public Domain Disk 1

Gsortc	This replaces Gsort ver.9 with version 12 to correct bugs
GShell32	Includes all of patches to update your original GShell. Includes a *MakeShell* Installation program

Graphics Public Domain Disk 3

Screen	3 screen savers "motion", "circle", "lines"

Telecommunications Public Domain Disk 3

Clock_Update	Clock modules that solve the IRQ problems

Patches Public Domain Disk 2

SmartWatch	Updated replacement for SpectroSystems Smart Watch Clock modules

Utility Public Domain Disk 7

ClrMem	Contains newer and better UNLINK and DEINIZ utilities. Reports status
Formats	Performs multiple formats and auto increments filename-number
UnLzh7	Latest version (bugs fixed) de-archiving utility
Cron (vers.10)	Task scheduler

Programming Public Domain Disk 3

CPrep	ANSI standard C preprocessor. Replaces Microware's **c.prep**

OS-9 Newsletter
In Next Months Issue

January always features our annual INDEX to all of the articles printed in the previous year's issues, so every listing, article, review, patch, project, etc., will be listed in the January issue. *In very small print!*

We are also looking forward to a review from the Longview Kelso CoCo group of the new FHL Kix30 OSK machine.

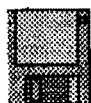BTW, Happy Birthday to us. We are now beginning our 4th year of publication.

OS-9 Calendar

March 27-28
Middle America's Fest
Des Moines, Iowa

June 18-19
PNW CoCo Fest
Port Orchard, Washington

Club Activities Report

Bellingham OS9 Users Group - Longview/Kelso CoCo Club
Mt. Rainier CoCo Club - Port O'CoCo Club - Seattle 68xxx Mug

Bellingham OS-9 Users Group

Once again our monthly meeting was canceled, this time due to the holiday vacation and the closing of the school where we have our meetings. However, like we did in November, we had a couple of mini meetings.

Our first meeting was brought on by an emergency. I crashed the hard drive on the GIMIX machine we have set up for student's usage at school. I called **Wes Payne** our GIMIX System Operator (SYSOP) and we got together on Wednesday evening, the 9th.

According to the addendum that came with the computers, it is not possible to make the computer boot from a hard drive, but obviously there must have been a more recent addendum because when I followed the procedure as outlined, the GIMIX booted strictly from the hard drive.

What was really neat was that the jumpers on the CPU card permitted the GIMIX to boot from a floppy after displaying a BIOS type menu similar to what is found on most MS-Dos machines. Everything was configured correctly so I continued the floppy boot up. Then I *OS9Gen*-ed a new boot file to the hard drive, *DSAVE*d everything else from the system floppy to the hard drive and then returned the jumper to it's original position. That was it!

Wes configured a new **STARTUP** file so that all 7 terminals were initialized properly and **TSMON** would force the standard **LOGIN** procedure. Wes then added his own special security trap for the Super User. We still have to install some of the major software, i.e., **Dynacalc** Spreadsheet, **Sculptor** Database, etc. **Dynacalc** is actually quite difficult to install because you must run a configuration program to create the **dynacalc.trm** file using information from the terminal's manual, so that **Dynacalc** will know how to read and write to your terminal. Didn't have to do that for the CoCo.

Our second meeting consisted of only myself, **Craig DuBois** and my wife **Barbara Alexander**. Our purpose was to touch up the audio track on the video tape we made last month showing how to install the 6309 chip in the CoCo. Craig has a special VCR machine that permits audio dubbing on a pre-recorded tape. Portions of the audio were clipped at the beginning of some of the scenes or camera position changes. We were able to record the audio right over the old audio track and make a smooth transition between scenes without any noticeable breaks or detection in the audio. Very cool!

-- Rodger Alexander --

The Bellingham OS-9 Users Group meets at Fairhaven

Middle School, 110 Parkridge Rd., Bellingham, WA. on the 4th Wednesday of each month at 7:00 p.m. Call (206) 734-5806 for more information.

◄─────────────────────────────►
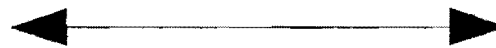
Mt. Rainier CoCo Club

The December meeting of Mt. Rainier Color Computer Club featured **Randy Kirschenmann**, demonstrating a demo program included with the *GUILIBIO.AR* file that has been recently released to all OS-9 Computer Network (**OCN**) region libraries and most OS-9 Bulletin Boards as well as Compuserv and Delphi OS-9 databases. It is a graphical interface C library with three dimensional capabilities for use by programmers. A rewrite was attempted however the resulting program would not compile. Tune in next month for the exciting conclusion.

Rick was having problems with getting his printer to print graphics so some time was spent determining which driver to use with his particular model printer. a (Star Gemini 10X).

Next month Randy will run a demo on desoldering a 6809 and installing the new 6309.

-- John Schliep --

The Mt. Rainier Color Computer Club meets the second Thursday of each month at the Fern Hill Library in Tacoma at 6:45 p.m.

◄─────────────────────────────►

Port O-CoCo Club

The last meeting of the year was almost a "pre-party" and a party. The pre-party was the last Computer Swap Meet of '92 in Kent. As usual, the CoCo Community has had a booth there to show off what our "littl' ol' game machine" can do. The backbone of the presentation were **Tom Brooks** (who had a serious day in the hospital just a week before but insisted on attending anyway), **Gene Elliott** (with our new CoCo in a tower), **Chris Johnson** (who will never miss a chance to show up any DOS machine) and **Terry Laraway** (who wooed more than a handful of the crowd with his **VIVID** color graphics.

The holiday crowd was in a great mood and many stopped by to look, chat and try to buy the unidentified machine in the handsome tower. It looked great, especially with its sturdy black soft grip handle on top. The Port O' CoCo Club purchased the SCUSI controller for our soon to come hard

drive for $15. We showed off our new CM-8 monitor that we purchased for only $49.95. Many of the CoCo people expressed interest in attending the upcoming party in Port Orchard.

The Computer Bank Charity had the table next to us. They had an unbelievable range of items looking for a loving home. The Charity was able to raise a record amount of money this year. Even 4 8" drives found a new home. The man was going to use the motors for his sons' robotics projects.

SPECIAL NOTICE: The CM-8s are down to $49.95 and they are hard to locate. The Port Orchard Radio Shack's manager, Norris Blair, offers a great arrangement. If you call him at (206) 876-5454 he will tell you the tax and shipping (if needed) for getting one. You send him a check for that amount. If he is unable to find a CM-8 via the RS computer communications system, which is updated daily, he will return your check. If he finds one for you, most likely when he finds one for you, you'll have a fine monitor at a criminally low price. And that's with the standard warranty! At this price you can think of a backup color monitor.

The third Monday of the month was Party Time! A record number of people showed up. And they were not empty handed. Food and drink filled one large table while computers covered two others. **Gene Elliott** brought our machine, which is looking better ever day, and **Terry Laraway** (the guy who just doesn't find time to get on the BBS PNW-OS9 echo and read all the hot news about himself and OS-9 for thousands of miles) brought his color ink jet printer and some of the GREAT clay paper that adds many degrees of pizzazz to all the colors. He handed out color prints to young and young in heart.

Before things got completely out of control, **Donald Zimmerman** conducted a brief meeting. The two major announcements were the upcoming Computer Fair at the Kitsap Mall in Silverdale on January 9th & 10th. Five other computer groups and 7 vendors will be participating in the event before thousands of shoppers that weekend. This is the 4th year we have participated in the 5 year old event and the second year we have organized it. This year we are striving to provide the vendors with a supply of flyers before the event to promote it and generate more interest. We are also going to clear a little money in our efforts. Anything left after printing costs will go to our club.

The second announcement was the A&M Computer Swap meet coming to Kitsap County. A&M is no longer providing free booth space to clubs and non-profits. It costs $50 to participant. We have been involve for about 2 1/2 half years. We will still be able to do a little promotion by having the Computer Bank Charity pass on the word about the CoCo community. Both **Chris Johnson** and **Donald Zimmerman** are board members of CBC and help with these events.

A new group is starting up computer swap meets in Tacoma and Olympia in 1993. They have offered free space to us. We will be taking advantage of that offer in February in an attempt to reach CoCo users in those respective areas.

Besides all the food, the highlight of the evening was the drawing. All of the 25 + attendees received one chance. Extras could be purchased. Items were from **EverSoft Ltd**, **Ted Jaeger** (of Fulton MO and author of DeskTamer 2.0), **Softline Computers** of Port Orchard, and **Northwest Computer Outlet** of Port Orchard. The children drew the numbers from a "No DOS" mug. Five people wound up with an extra gift under the tree!

Rodger Alexander, his wife **Barbara** and son **Rodger Jr.** drove all the way from Bellingham to participant. He brought along his HEAVY and LARGE GIMIX computer. The computer weighs in at 80 pounds. Also included were two terminals that Rodger plugged in at different ends of the room so that those of a curious nature could play around with OS-9 in it's native multi-user configuration.

The Party went on well past ten o'clock in spite of the next day being a work day for most. A great time was had by all and it was a great mix of computer and non-computer people. There were lots of floating from small group to small group. Of course, a morsel of something or other went down the hatch each time anyone passed the food table. Thanks to everyone for attending and bringing the goodies!
 -- Donald Zimmerman --

The Port O'CoCo Club meets the 3rd Monday of each month at 7 p.m. in the Stock Market Foods, half way up Mile Hill in Port Orchard.

◀   ▶

Seattle 68xxx MUG

The December meeting opened with a video tape presentation by **Rodger Alexander** demonstrating how to piggy back a Hitachi 6309 cpu chip on top of the existing 6809 in the CoCo-3.

Scott Honaker was recently back from COMDEX in Las Vegas and reported to us some of the more impressive displays one of which was the very, very small ROM Pak type devices the size of credit cards that plug into small lap top computers. Would you believe a 20Meg Hard Drive that fits in my wallet!

Another item was the high speed graphic cards using the 32bit bus slot on some of the newer machines that run at the cpu clock speeds. That means a Super VGA graphics accelerator card was able to provide graphics displays several times faster than the 50MHz clock speed. Made super high resolution graphics display as quick or quicker than standard text screen displays.

There was also an IDE interface/controller card that was designed to plug into the "super" slot permitting a large media IDE drive to get and put data from and to the computer almost as fast as a ram drive.

The January meeting will feature a presentation of Packet Radio and other ways Ham's can use the CoCo for Radio Telecommunications.
 -- Rodger Alexander --

Washington State BBS List

COLUMBIA HTS. BBS
-- Lonview/Kelso --
RiBBS (FidoNET)
(206) 425-5804

DATA WAREHOUSE BBS
-- Spokane --
RiBBS (FidoNET)
(509) 325-6787

BARBEQUED RIBBS
-- Bellingham --
PC-Board (PC-Net) - CoCo Conference #5
(206) 676-5787

OS-9 TACOMA BBS
-- Tacoma --
RiBBS (FidoNET)
(206) 566-8857

ULTIMATE EXPERIENCE BBS
-- Anacortes --
RiBBS (MaxNET)
(206) 299-0491

OS-9 Newsletter
3404 Illinois Lane
Bellingham, WA 98226-4238