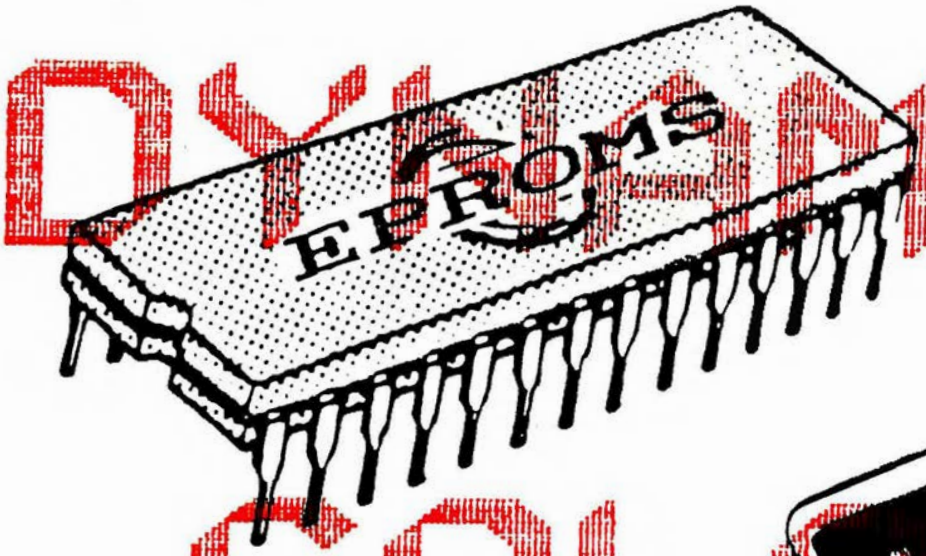
RADIO SHACK COLOR COMPUTER MAGAZINE

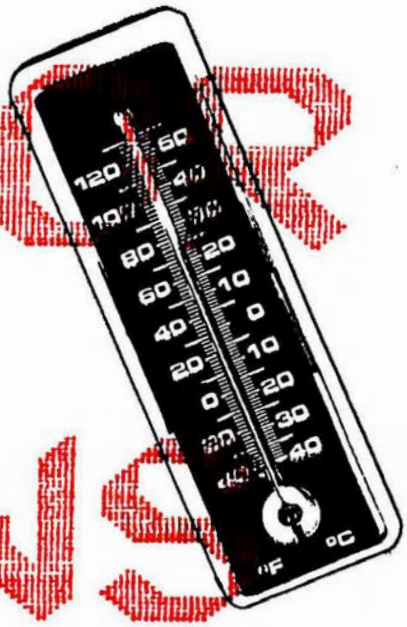
April 1987

Vol. 4 No. 3

$1.95



DYNACOM COLOR NEWS



Measuring Temperature
Ham Radio
ML Programming
Color Computer 3
+ Much more

DYNAMIC COLOR NEWS is published monthly by DYNAMIC ELECTRONICS, INC., P.O. Box 896, Hartselle, AL 35640, phone (205) 773-2758. Bill Chapple, BA, BSE President; Dean Chapple, Sec. & Treas. ; John Pearson, Ph. D. Consultant; Bob Morgan, Ph. D., Consultant.

 Subscriptions are $15/yr for U.S.A. $18 Canada & Mexico, $30 other foreign.

The purpose of this magazine is to provide instruction on Basic & Machine Language programming, Computer theory, operating techniques, computer expansion, plus provide answers to questions from our subscribers.

The submission of questions, operating hints, and solutions to problems to be published in this magazine are encouraged. All submissions become the property of Dynamic Electronics if the material is used. We reserve the right to edit all material used and not to use material which we determine is unsuited for publication.

We encourage the submission of Basic and Machine Language Programs as well as articles. All Programs must be well documented so the readers can understand how the program works. We will pay for programs and articles based upon their value to the magazine. Material sent will not be returned unless return postage is included. Basic & ML programs should be sent on a tape or disk & comments should be sent as a DAT or BIN file.

DYNAMIC COLOR NEWS

April 1987

Editor and Publisher
Bill Chapple W4GQC

Secretary
Dean Chapple

CONTENTS

256K & 512K MEMORY UPGRADES

We are closing out these **Banker** RAMS by J & R Electronics. These upgrade the older **D, E, F or 285** and earlier CoCo 2 computers with **4164** memory chips and a socketed SAM (6883) chip. Features include:

* Fast 35/40 Track Ramdisk (2 Ramdisks with 512K)
* 32K to 200K printer spooler (400K with 512K Ram)
* More than 30 PMODE 4 screens
* OS-9 Ramdisk 35-40 track single sided or 40 track double sided with 512K
* Memory protected when reset
* Toggle switch for 64K mode
* Compatible with all software
* **Complete ready to install**

ME-16- 256K RAM $79.95
ME-16A- 512K RAM 139.95

KOREAN CC-2 256K UPGRADE

Two plug in assemblies will upgrade the new CoCo-2 computers to 256K. Two miniature toggle switches allow independently selecting any one of the 4-64K memories. Features include:

* Powerful Memory Manager software allows maximum use of each bank. Use the ramdisk or the second 32K bank.
* Load any combination of programs into the banks. Quickly switch from one to the other.
* Easy solderless installation requires drilling two small holes for the switches.

Order ME-18 $99.95

128K MEMORIES

Same as the ME-18 except has one switch and 2-64K memories with Memory Manager Software. Upgrades the new CoCo-2 Computers to 128K. **Order ME-10A $39.95**

Upgrade 8-chip 4164 type 64K computers to 128K with Memory Manager software. **ME-12 $39.95**

MEMORY MANAGER - A complete set of software for managing the second 32K memory bank in 64K or larger computers. Run Basic programs in each bank or use the Ramdisk for program storage. Available free with our memory upgrades. $21.95 Disk or Tape.

MEMORY SAVER

Battery backup prevents loss of programs due to power failures. Mounts under keyboard. Consists of dry recharageable battery, control circuit, & miniature toggle switch. Will power a color computer for a couple of hours during a power failure. For CC-2 & older computers. Price reduced. $39.95

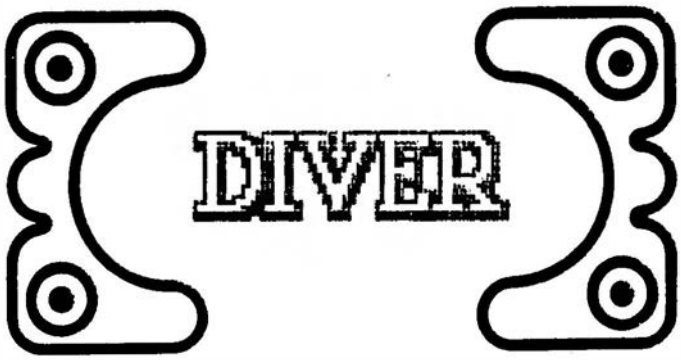
DISTO RAMDISK

A 256K Ram that plugs into a slot on a Multi-Pak expander. Works with all color computers. Copy a disk into the Ram or make multiple copies from the Ram to disks. No modifications to the computer are required. Software is included **$119.95.**

Dyterm -Terminal Program **$9.95**
Disasm-Decimal Assembler **$9.95**

Checks, VISA & MC Cards
Add $3 Shipping

DYNAMIC ELECTRONICS Inc.
BOX 696 (205) 772-2756
HEADLAND, AL 36345



The object of Diver is to carefully manuever your minisub around the depths of the ocean. Try to avoid the bubbles and other obstacles. You must reach the treasure chest at the bottom ant then surface. Use the right joystick to control the game action. This program is provided by T & D Software (see their advertisement on page 8) and is used by permission.

```
Ø REM COPYRIGHT (C) T&D SOFTWARE
    1987          diver
1 PMODEØ:GOTO6ØØØØ
2 REM *          DIVER          *
3 REM *  BY CHRISTIAN KEYES *
4 REM *       FEB. 1986         *
5 REM ************************
6 PMODE3,1:PCLS3:PMODE4,1:CLS:PR
  INT@1Ø9,"diver":PRINT:PRINT"
  DIVE FOR TREASURE AND RETURN
     IT TO THE SURFACE FOR POIN
  TS.   CAREFULLY CONTROL THE P
  ROBE     WITH THE RIGHT JOYST
  ICK.
7 PRINT:INPUT"   USE THE SPEED U
  P POKE";A$:IFLEFT$(A$,1)="Y"T
  HENINPUT"       IS THIS A COCO
   3";A$:IFLEFT$(A$,1)="Y"THENP
  OKE65497,ØELSEPOKE65495,Ø
1Ø DIM L$(26),DS(12,6),N$(1Ø),BB
  (12,6)
11 N$(Ø)="BUU4NF4ER3FD4GL3BR7":N
  $(1)="BR3RU6NGD6R2BR3":N$(2)=
  "BU5ER3FDG2L2GDR5BR3":N$(3)="
  BU5ER3FDGNLFDGL3HBDBR8":N$(4)
  ="BU6D3R4NU3NRD3BR4":N$(5)="B
  U6NR5D2R4FD2GL3HBDBR8":N$(6)=
  "BUU4ER3FBD2BLNL3FDGL3BR7":N$
  (7)="BU6R5DG4DBR7"
12 N$(8)="BUUEHUER3FDGNL2FDGL3BR
  7":N$(9)="BUFR3EU4HL3GDFR4BD3
  BR3":L$(1)="U4E2F2D2NL4D2BR4"
  :L$(2)="NR3U6R3FDGNL3FDGBR5":
   L$(3)="BUFNR3HU4ER3BD6BR4":L$
  (4)="NR3U6R3FD4GBR5":L$(5)="N
  R4U3NR3U3R4BD6BR4":L$(6)="U3N
  R3U3R4BD6BR4"
13 L$(7)="BUFR2EU2L2BL2D3U5ER2FB
  D5BR4":L$(8)="U6D3R4U3D6BR4":
  L$(9)="R4L2U6L2R4BD6BR4":L$(1
  Ø)="BUFR2EU5BD6BR4":L$(11)="U
  6D3RNE3F3BR4":L$(12)="NU6R4BR
  4":L$(13)="U6F2E2D6BR4":L$(14
  )="U6DF4DNU6BR4":L$(15)="BUU4
  ER2FD4GL2HFBR7":L$(16)="U6R3F
  DGL2BD3BR7"
14 L$(17)="BUU4ER2FD4GL2HFR2EHF2
  LBR4":L$(18)="U6R3FDGL3R2F2DB
  R4":L$(19)="BUFR2EUHL2HUER2FB
  D5BR4":L$(2Ø)="BU6R4L2D6BR6":
  L$(21)="BU6D5FR2EU5D6BR4":L$(
  22)="BU6D2FD2FEU2EU2BD6BR4":L
  $(23)="NU6E2F2NU6BR4":L$(24)=
  "UE4UBL4DF4DBR4"
15 L$(25)="BU5UDF2E2UDG2D3BR6":L
  $(26)="BU6R4DG4D1R4BR4"
16 GOTO69
17 PMODE4,1:PCLS:SCREEN1,1
18 TR=Ø:HI=Ø:L=1:SC=Ø:YM=1:PLAY"
  L255T255V31":XM=Ø
19 POKE178,2:LINE(Ø,2Ø)-(3Ø,Ø),P
  SET,BF
2Ø POKE178,3:LINE(2,2)-(14,8),PS
  ET,BF
21 POKE178,Ø:DRAW"BM2,2S8R6D3L6U
  3BR2BD1R2BDL2":POKE178,2
22 GET(2,2)-(14,8),DS,G
23 PCLS
24 LINE(Ø,5Ø)-(255,191),PSET,BF
25 GET(128,96)-(14Ø,1Ø2),BB,G
26 POKE178,3:DRAW"BM124,26S2R8D1
  2R16NL4ØD8R28NL96M+4,+12R24NL
  156M-8,+16L14ØM-8,-16R24M+4,-
  12R28U8R16U12R2":CIRCLE(1Ø4,4
  Ø),2:CIRCLE(124,4Ø),2:CIRCLE(
  144,4Ø),2
27 CIRCLE(126,22),1:CIRCLE(125,2
  Ø),2:CIRCLE(124,18),2:CIRCLE(
  123,16),4
28 POKE178,1:PAINT(124,27),,1:PA
  INT(124,34),,1:PAINT(12Ø,38),
  ,1:PAINT(124,44),,1
```

```
29 POKE178,Ø:LINE(199,5)-(255,15
   ),PSET,BF:POKE178,3:DRAW"S4BM
   2ØØ,1Ø;XL$(12);XL$(5);XL$(22)
   ;XL$(5);XL$(12);":L$=STR$(L):
   IFL>9THENDRAWN$((INT(L/1Ø)))
3Ø DRAWN$(VAL(RIGHT$(L$,1))):DRA
   W"CØBM12Ø,187S2D12R4ØU12NL4ØM
   -4,-5L32M-4,+5"
31 DRAW"C5BM1Ø,1ØS4;XL$(19);XL$(
   3);XL$(15);XL$(18);XL$(5);BL1
   BUUBU2UBD5BR4":GOSUB118
32 PMODE3,1
33 FORX=1 TO 25
34 BX=RND(255)
35 BY=RND(191):IF BY<62 OR BY>17
   Ø THEN 35 ELSE 36
36 DRAW"C8":CIRCLE(BX,BY),1:CIRC
   LE(BX,BY),2:CIRCLE(BX,BY),3:C
   IRCLE(BX,BY),4
37 PLAY"O5B"
38 NEXTX
39 PMODE4,1
4Ø PUT(118,51)-(13Ø,57),DS,PSET
41 XC=118:YC=51
42 X=SGN(INT((JOYSTK(Ø)-5)/58)):
   XM=XM+X:IFXM<-3THENXM=-3ELSEI
   FXM>3THENXM=3
43 Y=SGN(INT((JOYSTK(1)-5)/58)):
   YM=YM+Y:IFYM>3THENYM=3ELSEIFY
   M<-3THENYM=-3
44 IFXC+XM>242ORXC+XM<1THENXM=Ø:
   GOTO42
45 IFYC+YM<52ANDTR=1THENGOTO96
46 IFYC+YM>184ORYC+YM<52THENYM=Ø
   :GOTO42
47 PMODE3,1
48 IFYM=ØTHEN55
49 IFYM<ØTHENW=ØELSEW=7
52 FORCX=XC+XM TO XC+XM+12
53 IFPPOINT(CX,YC+W+YM)=8THENHI=
   5:GOTO62
54 NEXTCX
55 IFXM=ØTHEN62
56 IFXM<ØTHENW=ØELSEW=13
58 FORCY=YC+YM TO YC+YM+6
6Ø IFPPOINT(XC+W+XM,CY)=8THENHI=
   5:GOTO62
61 NEXTCY
62 PMODE4,1
63 POKE178,2:LINE(XC,YC)-(XC+12,
   YC+6),PSET,BF:POKE178,3:XC=XC
   +XM:YC=YC+YM:PLAY"A"
64 PUT(XC,YC)-(XC+12,YC+6),DS,PS
   ET:PLAY"G"
65 IFHI=5THENGOTO85
66 IFYC>175ANDTR=ØTHEN67ELSE68
67 IFXC>11ØANDXC<139THENGOTO1Ø8
68 GOTO42
69 PMODE4,1:PCLS:SCREEN1,1
7Ø FORX=5TO25ØSTEP8:CIRCLE(X,5),
   5:CIRCLE(X,186),5:NEXTX
71 FORX=5TO186STEP8:CIRCLE(6,X),
   5:CIRCLE(25Ø,X),5:NEXTX
72 DRAW"BM6Ø,5ØS2ØNR3U6R3FD4GBR"
   :DRAW"BM65,43S1ØNR3U6R3FD4GBR
   "
73 DRAW"BM88,2ØS2ØR2D6L2U6"
74 DRAW"BM1Ø3,2ØS2ØR2M+1,+3M+1,-
   3R2M-3,+6M-3,-6"
75 DRAW"BM137,2ØS1ØR8D2L6D3R3D3L
   3D3R6D2L8U12"
76 DRAW"BM163,2ØS1ØR8D6L5M+5,+6D
   1L2M-4,-5D5L2UM163,2ØBM+2,+2R
   4D2L4U2"
77 POKE178,2:PAINT(62,48),,1:PAI
   NT(9Ø,22),,1:PAINT(1Ø5,22),,1
   :PAINT(139,22),,1:PAINT(165,2
   2),,1:POKE178,3
78 DRAW"BM11Ø,9ØS8;XL$(2);XL$(25
   );"
79 DRAW"BM5Ø,11ØS8;XL$(3);XL$(8)
   ;XL$(18);XL$(9);XL$(19);XL$(2
   Ø);XL$(9);XL$(1);XL$(14);"
8Ø DRAW"BM82,13ØS8;XL$(11);XL$(5
   );XL$(25);XL$(5);XL$(19);"
81 'DRAW"BM3Ø,17ØS4;XL$(16);XL$(
   18);XL$(5);XL$(19);XL$(19);BR
   1Ø;XL$(5);XL$(14);XL$(2Ø);XL$
   (5);XL$(18);BR1Ø;XL$(2Ø);XL$(
   15);BR1Ø;XL$(3);XL$(15);XL$(1
   4);XL$(2Ø);XL$(9);XL$(14);XL$
   (21);XL$(5);"
82 FORX=1TO31:PLAY"L255T255O5V"+
   STR$(X)+"CDEFGAB":NEXTX
84 GOTO17
85 FORXX=1TO2Ø:PLAY"L255T255V31O
   5G":POKE178,2:LINE(XC,YC)-(XC
   +12,YC+6),PSET,BF:PLAY"G":PUT
   (XC,YC)-(XC+12,YC+6),DS,PSET:
   NEXTXX
86 PLAY"L1T9V2ØO1BAGFEDC"
87 POKE178,Ø:LINE(64,64)-(191,12
   7),PSET,BF
88 POKE178,3:DRAW"BM68,75;XL$(25
   );XL$(15);XL$(21);BR4;XL$(8);
   XL$(1);XL$(22);XL$(5);BR4;XL$
   (3);XL$(18);XL$(1);XL$(19);XL
   $(8);XL$(5);XL$(4);R"
89 DRAW"BM98,85;XL$(25);XL$(15);
   XL$(21);BR4;XL$(12);XL$(15);X
   L$(19);XL$(5);R"
9Ø DRAW"BM76,1Ø5;XL$(16);XL$(18)
   ;XL$(5);XL$(19);XL$(19);BR4;X
   L$(1);BR4;XL$(11);XL$(5);XL$(
   25);BR4;XL$(2Ø);XL$(15);"
91 DRAW"BM86,115;XL$(16);XL$(12)
   ;XL$(1);XL$(25);BR4;XL$(1);XL
   $(7);XL$(1);XL$(9);XL$(14);R"
```

```
92 A$=""
93 A$=INKEY$:EXEC43345:IFA$=""THEN93
94 PMODE3,1:PCLS3
95 GOTO17
96 FORGH=1TO31STEP2:PLAY"L255T255O2V"+STR$(GH)+"CDEFGABBAGFEDC":NEXTGH:FORGH=31TO1STEP-2:PLAY"L255T255O2V"+STR$(GH)+"CDEFGABBAGFEDC":NEXTGH
97 POKE178,0:LINE(0,175)-(255,191),PSET,BF:POKE178,3
98 DRAW"BM12,182;XL$(25);XL$(15);XL$(21);BR4;XL$(8);XL$(1);XL$(22);XL$(5);BR4;XL$(13);XL$(1);XL$(4);XL$(5);BR4;XL$(9);XL$(20);BR4;XL$(16);XL$(1);XL$(19);XL$(20);BR4;XL$(20);XL$(8);XL$(9);XL$(19);BR4;XL$(12);XL$(5);XL$(22);XL$(5);XL$(12);R"
99 DRAW"BM4,193S4;XL$(25);XL$(15);XL$(21);BR4;XL$(14);XL$(15);XL$(23);BR4;XL$(16);XL$(18);XL$(15);XL$(7);XL$(18);XL$(5);XL$(19);XL$(19);BR4;XL$(20);XL$(15);BR4;XL$(20);XL$(8);XL$(5);BR4;XL$(14);XL$(5);XL$(24);XL$(20);BR4;XL$(12);XL$(5);XL$(22);"
100 DRAW"XL$(5);XL$(12);R"
101 PLAY"V31L10T2O3BDAEGFGEADBC"
102 FORXD=1TO800:NEXTXD
103 POKE178,2:LINE(0,175)-(255,191),PSET,BF
104 SC=SC+(L*25*100):GOSUB118
105 TR=0:L=L+1:XM=0:YM=1:PLAY"L255T255V31"
106 POKE178,2:LINE(XC,YC)-(XC+12,YC+6),PSET,BF:POKE178,3
107 GOTO29
108 FORD=1TO31:PLAY"V"+STR$(D)+"L255T255O5BO1AO5B":NEXTD
109 POKE178,0:LINE(0,175)-(255,191),PSET,BF
110 DRAW"C5BM28,182S4;XL$(20);XL$(8);XL$(5);BR4;XL$(20);XL$(18);XL$(5);XL$(1);XL$(19);XL$(21);XL$(18);XL$(5);BR4;XL$(8);XL$(1);XL$(19);BR4;XL$(2);XL$(5);XL$(5);XL$(14);BR4;XL$(20);XL$(1);XL$(11);XL$(5);XL$(14);R"
111 DRAW"BM18,193S4;XL$(14);XL$(15);XL$(23);BR4;XL$(7);XL$(5);XL$(20);BR4;XL$(2);XL$(1);XL$(3);XL$(11);BR4;XL$(21);XL$(16);BR4;XL$(20);XL$(15);BR4;XL$(20);XL$(8);XL$(5);BR4;XL$(19);XL$(21);XL$(18);XL$(6);XL$(1);XL$(3);XL$(5);R"
112 FORD=31TO1STEP-1:PLAY"V"+STR$(D)+"L255T255O5BO1AO5B":NEXTD
113 FORD=1TO800:NEXTD
114 POKE178,2:LINE(0,175)-(255,191),PSET,BF
115 PUT(XC,YC)-(XC+12,YC+6),DS,PSET
116 XM=0:YM=-1:PLAY"V31":TR=1:SC=SC+(250*L):GOSUB118
117 GOTO42
118 SC$=STR$(SC):POKE178,0:LINE(50,4)-(85,10),PSET,BF:POKE178,3:DRAW"C5BM52,10;":FORS=1TOLEN(SC$):DRAWN$(VAL(MID$(SC$,S,1))):NEXTS:RETURN
60000 PCLEAR4:GOTO2
```

Last month we presented some material on eraseable programable read only memories (EPROMS). We discussed the two types of EPROMS and their differences in the programming voltages. The earlier EPROMS required about 21 volts and the newer ones require about 12.5 volts.

What can be done with an EPROM? An EPROM can be used to replace the basic and extended basic chips in the computer. EPROMS can be placed in cartridges to run games or basic programs. Placing programs in a cartridge is not a problem unless a disk drive is being used because the disk drive uses the same memory area as a cartridge. However the cartridge can be relocated to upper memory by an integrated circuit. A "Y" cable

can be used to allow both the disk drive and the cartridge to operate together. If you do not have a disk system, an EPROM can be much more useful for storing programs due to the slower speed of tape.

Since there is about 8K of upper memory available, how can EPROMS larger that 8K be used in a cartridge? Memory blocks of 8K can be switched by electronics or with switches. The switch method is easy to use and we will show how to implement it.

Let's look at the 27128 EPROM. This contains 16K bytes or two blocks of 8K bytes. A pin diagram is shown in figure 1. A single pole double throw switch can used used to select the banks. Connect the center of the switch to pin 26 which is A13. The other two terminals of the switch should be tied to +5V (VCC) and GND.

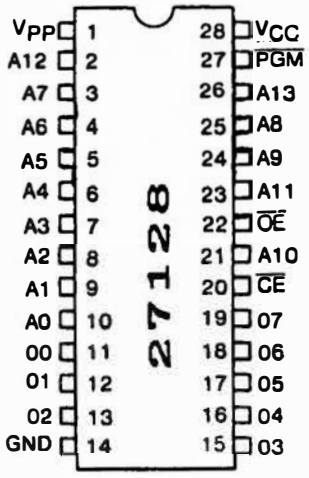


FIGURE 1

BACK ISSUES

Back issues of **DYNAMIC COLOR NEWS** are available for **$1.95** each, **3** for **$5**, or **12 for $15 pp**.

Foreigners other than Canada add **$2** for Air Mail postage.

For a 27256 EPROM two switches can be used to select the 4-8K memory banks if they are to be used in the upper 8K memory location. Tie the centers of the switches to A13 and A14. These will be pins 26 and 27. Connect the other two terminals to VCC and GND. A Diagram of the 27256 is shown in Figure 2. Note that it is identical to the 27128 except for pin 27. This is A14 for the 27256.

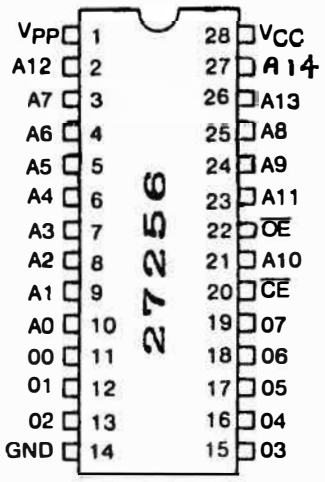


FIGURE 2

USING the UPPER 8K

A "Y" cable can be used to allow a cartridge and a disk drive to be used together. However some circuitry is required to force the EPROM to use the upper 8K of memory. The disk drive uses memory from 49152 ($C000) to 57343 ($DFFF) and the upper 8K is from 57344 ($E000) to 65535 ($FFFF). However the upper 256 bytes are reserved for the system and can not be used for programs. Let's make a chart of the most significant address lines.

32K	16K	8K	
A15	A14	A13	Value
1	1	1	$E000
1	1	0	C000
1	0	1	A000
1	0	0	8000

SPECTROGRAM

MAGAZINE *FOR COLOR COMPUTER USERS.*

* **Programs for business, home management, self-improvement, games, and utilities**
* **Reviews of Color Computer products**
* **Tutorials on programming in Assembly, C, Pascal, and Basic**
* **Contests**

As an introductory offer, you can order the first year of SPECTROGRAM Magazine at 40% off the cover price. For $18, you will receive 12 issues of the magazine that could become the most informative addition to your Color Computer system.

We want to establish a line of two-way communication between our staff and our readers as an aid in serving your needs. Please enclose any comments or special requests with your subscription form.

GROUP RATES: $15 with orders of five or more subscriptions!

PLEASE SEND ME 12 ISSUES OF SPECTROGRAM MAGAZINE FOR $18 ***(40% OFF THE COVER PRICE).***

Name: _____

Address: _____

City: _____

State: _____ **Zip:** _____

( ) Check Enclosed ( ) Visa ( ) Mastercard

Card # _____ **Exp. Date:** _____

Mail to: **SPECTROGRAM Magazine**
P.O. Box 138
Rockford, IL 61105
(815)968-9600

COMPUTER TYPE:
( ) 64K Color Computer
( ) 32K Color Computer
( ) 16K Color Computer
( ) 4K Color Computer
( ) Other--Specify _____

PERIPHERALS:
( ) Printer Type _____
( ) Modem Type _____
( ) Disk Drive (0) (1) (2) (3)
( ) Multi-Pak Interface
( ) Other--Specify _____

LANGUAGES:
( ) Extended Basic
( ) Color Basic
( ) Disk Basic
( ) Basic09
( ) Pascal
( ) C Compiler

PROGRAM PREFERENCE:
( ) Business
( ) Games
( ) Graphics
( ) Tutorials
( ) Utilities
( ) Home Management
( ) Self-Improvement

SPECIAL DEAL ON 500 PROGRAMS!

GET 50 DISKS OR 50 CASSETTE TAPES FULL OF OVER 500 PROGRAMS. HERE IS WHAT YOU'LL RECEIVE:

* Over 250 Utility/Home Application Programs including a Word Processor, DataBase, Spreadsheet, Account Manager, 2 Basic Compilers, Terminal Programs, ROM Copies, Mail List, Machine Language Tutorials, Plus Much More!
* Over 200 exciting games including Warlords, Star Trek, Super Vaders, Solar Conquest, Horse Races, Football, Baseball, Frog Jump, Invader, Plus Much More! (Many machine language games)
* Over 30 adventures including The College Adventure, Dungeon Master, Space Lab, Ice World, Ship Wreck, Zigma Experiment. Plus 32K Graphic Adventures.

EACH INDIVIDUAL ISSUE SOLD FOR $9.00 EACH OR $450 FOR ALL 50 ISSUES. WE SLASHED THE PRICE TO ONLY 149.99.

REG. $450 NOW $149.99

★★THIS MONTH ONLY★★

Buy this package of 500 programs and receive a free 6 month subscription. (A $35 value)

VISA

RAINBOW CERTIFICATION SEAL

THE GREATEST SOFTWARE DEAL ON EARTH JUST GOT BETTER!

THAT'S RIGHT! THIS MONTH WE'VE DROPPED OUR YEARLY SUBSCRIPTION RATE AN UNBELIEVABLE $10.00 TO ENTICE YOU INTO SUBSCRIBING WITH US. GET 12 DISKS OR TAPES A YEAR CONTAINING OVER 120 QUALITY PROGRAMS. A SUBSCRIPTION TO T & D SOFTWARE CONSISTS OF 10 READY-TO-LOAD PROGRAMS DELIVERED BY FIRST CLASS MAIL EVERY MONTH.

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

PRICES

	TAPE OR DISK	THIS MONTH ONLY
1 YEAR (12 issues)	~~70.00~~	60.00
6 MO. (6 issues)	~~40.00~~	35.00
1 ISSUE	~~9.00~~	6.00

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

* 16K-64K Color Computer
* Over 4000 Satisfied Customers
* Back Issues Available From
* July '82 (Over 500 Programs)

OUR LATEST ISSUE CONTAINED

1. Accounts Receivable
2. Work Mate
3. Calendar
4. Invasion
5. Trip Adventure
6. Foot Race
7. Flippy the Seal
8. Screen Calculator
9. Able Builders
10. Super Error 2

RAINBOW CERTIFICATION SEAL

Available on COCO 1, 2 and 3!
All Programs Include Documentation!

MasterCard

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

TABLE 1

We assumed that all lower order addresses are Ø. If the upper 8K is addressed then we want to disable the disk controller at $CØØØ. Notice that A13 is a Ø for CØØØ and a 1 for EØØØ. A13 can be used to switch the disk controller off and the EPROM on.

When the upper 16K of memory is selected pin 32 of the cartridge port goes low. Let's call this signal CA. We can use this signal along with A13 of the address bus to select the disk controller or the EPROM. Let's look at the following table:

CA	A13	Disk	EPROM
Ø	Ø	Ø	1
Ø	1	1	Ø

TABLE 2

Computer support chips require the control line to go low or to a logic Ø to enable the chip. Notice that CA and A13 are the only signals needed to enable either the disk controller or the EPROM.

HARDWARE

One of the easiest chips to use for simple logic is a 1 of 1Ø or 1 of 8 decoder. These transistor transistor logic (TTL) chips have outputs that go low when they are selected by an input address. Let's look at a truth table for the 74LS42 which is a 1 of 1Ø decoder.

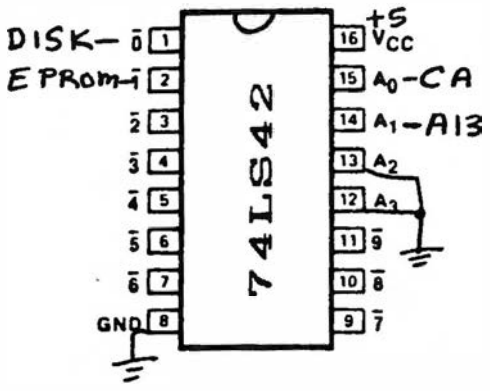


FIGURE 3

12	13	14	15	Address pins
A3	A2	A1	AØ	OUTPUT
Ø	Ø	Ø	Ø	Pin 1 low
Ø	Ø	Ø	1	Pin 2 low
Ø	Ø	1	Ø	Pin 3 low
Ø	Ø	1	1	Pin 4 low
Ø	1	Ø	Ø	Pin 5 low
Ø	1	Ø	1	Pin 6 low
Ø	1	1	Ø	Pin 7 low
Ø	1	1	1	pin 9 low
1	Ø	Ø	1	pin 1Ø low
1	Ø	1	Ø	pin 11 low

TABLE 3

Notice that the first two conditions in Table 3 meet our requirements. There are 1Ø outputs for the 74LS42, but we will only need to use two of them. We need to connect A3 and A2 to ground or logic Ø. Then we can connect CA to A1 and A13 of the address bus to AØ of the 74LS42. A wiring diagram of the 74LS42 is shown in Figure 3.

HARDWARE OPTIONS

There are two possibilities for installing an EPROM for programs. The first involves installing the EPROM inside the computer. This is what we did with our 96KX modules. However since the newer computers have most of the chips soldered in, this is not as easy to do as it was with the older computers. Fortunately the signals needed for an external EPROM are contained within the Basic and Extended basic ROM chips. A socket can be soldered to the top of these chips to hold the EPROM. Then it is a matter of doing some internal wiring within the computer. This method allows the EPROM to become a permanent part of the computer. There are 3 programs that we use frequently. The first is the Telewriter word processor. The second is our DYTERM program for downloading files from our model 1ØØ. The

third is our DYTERM assembler. If these programs were in EPROMS then they could be run from the EPROMS or quickly loaded into the computer.

When the EPROM is selected by addressing the upper memory, the cartridge port has to be deselected. This means the line going to pin 32 of the expansion port has to be broken. Pin 1 of the 74LS42 will go to pin 32 of the expansion port. The other side of the cut is CA and goes to pin 15 of the 74LS42.

Next month we will continue and give wiring details for mounting EPROMS inside the computer. Also we will show how to modify a "Y" cable to use an EPROM cartridge with a disk drive. There is much more material we can cover on using EPROMS. They are not very expensive and can be used for storing basic and machine language programs.

6809 PROGRAMMING

BY

JOHN BELLO

PART #12

INDEXING AND TABLE HANDLING

An important concept to grasp for the Assembly language programmer is Index addressing. Index addressing allows you to vary the address of the data you are accessing by changing the contents of a register. We do this by loading one of the 16 bit index registers X, Y or U with the starting address of the memory area or data table we want to access or manipulate. Then we "offset" this starting address to create what is called the "effective" address. This displacement or offset can be either in a positive or negative direction. The following are examples:

```
LDB -3,X
LDB -2,X
LDB -1,X
LDB  Ø,X
LDB +1,X
LDB +2,X
LDB +3,X
```

The 68Ø9's has special addressing modes called autoincrement and autodecrement to make indexing easier. The autoincrementing mode is denoted by the "+" symbol in the operand field location after the index register we use. This instruction automatically adds one to the index register after being executed. We could also, autoincrement the index register by two. This is signified by using two "++" symbols. The index register may also be autodecrenented by one or two. However in this case it is important to remember that the index register is decremented by one or two "before" the instruction is executed.

```
LDA ,X+     ;AUTO INC AFTER
            ;LOADING A REGISTER

LDA ,X++    ;AUTO INC X BY TWO
            ;AFTER LOADING A


LDA ,-X     ;AUTO DEC X ONE
            ;BEFORE LOADING A

LDA ,--X    ;AUTO DEC X BY TWO
            ;BEFORE LOADING A
```

NOTE THE PLACEMENT OF THE + AND - SYMBOLS.

Using index addressing we can process large blocks of data. Here is a routine that places the letter A on the entire video screen.

```
START   LDX #$400 ;POINT TO BASE
        LDA #'A   ;LOAD WITH LET
        LDB #511  ;COUNT
LOOP    STA ,X+   ;PUT ON SCREEN
        DECB      ;ONE LESS
        BNE LOOP  ;NOT END LOOP
        SWI       ;FINISHED
        END
```

In the example we first loaded the X index register with the address that points to the video screen, where we want to put the data. Then we load the A register with the value of the letter A and then load B with 511 which is the number of times we will loop through the routine. At LOOP we store the value in the A register at the address pointed to by the X register, which is autoincremented after the letter is stored on the screen. We then decrement the count held in the B register by one and test to see if the value in the B register has been made zero by the decrement . If it is not zero the BNE (Branch not Equal to Zero) command sends the program back to LOOP to do it another time. If the B register does equal zero, the program falls through to the SWI command that ends the routine.

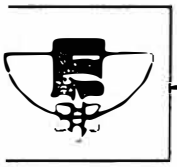
When we move forward using the autoincrement mode we start the index register at the first memory address. When moving backwards using autodecrement, start the index register one memory address beyond the highest address in the area we wish to access. We could use indexing to print a message on our video screen. To print a character we will use a ROM routine located a $A3ØA.

```
START   LDX  #MES    ;POINT TO
                     ;MESSAGE
PRT     LDA  ,X+     ;GET A CHAR
        BEQ  RET     ;END?
        JSR  $A3ØA   ;PRINT IT
        BRA  PRT
RET     SWI          ;FINISHED
MES     FCC  /DYNAMIC COLOR/
        FCB  Ø
        END
```

In the preceding program we point the X register to the message we want to print. Then we load the A register with a character and autoincrement the X index register. If the character loaded is zero, then we have finished printing. If it is not, we print the character now in the A register using the ROM subroutine and loop back to do another. It's important to place a zero after any message you wish to print using this routine or you might get caught in an endless loop since the routine will continue to print

RENEWAL TIME?

The date beside your name on the address label indcates the last issue you will receive. Send in your renewal if you want to continue receiving technical information on Color Computers. This is the last issue for those with 4/87 .



EMERSON COMPUTER SERVICES CO.

8289 BANNER ROAD S.E. • PORT ORCHARD, WASHINGTON 98366 • (206) 857-7878

SINCE 1983, E C S HAS BEEN THE NORTHWEST'S ONLY COMPUTER CONSULTING FIRM DEVOTED SOLELY TO THE COLOR COMPUTER AND ITS APPLICATIONS FOR *BUSINESS* AND PRACTICAL HOME USES.

PROGRAMS

JULIE THE MOUSE VER. 3.2
by
STEPHEN CLARK, CDP

EXTREMELY HANDY, VERSATILE
FREE-FORM DISK FILE MANAGER
MOUSE/JOYSTICK CONTROLLED.
(see full page ad in
October, November, December)
Only $21.95 Shipping included

****

ALL BOB van der POEL SOFTWARE

ULTRA-TELEPATCH (updated 12- '86)
NOW WITH HANGING INDENT!!!
$19.95 + $1.50 shipping

NEW FOR COCO III !!!
CRIBBAGE GAME
Extremely Challenging
Lots of new CC III graphics
$13.95 + $1.50 shippinng

NEW FOR COCO III PROGRAMMERS !!!
ULTRA-EDITOR * 2 BUFFERS !
16K & 50K copy between them!
MUCH, MUCH MORE !!!! $19.95

MAGAZINE INDEX SYSTEM
$14.95 + $1.50 shipping

****
ALL WORKBASE DATA SYSTEMS SOFTWARE

DATABASE MANAGERS WORKBASE I AND II
EXTREMELY POWERFUL RELATIONAL DBMS
DESIGIGNED FOR BUSINESS USERS.
$79.95

ALL WORKBASE BUSINESS PORTFOLIOS

STAND ALONE MENU DRIVEN ACCOUNTING
SOFTWARE THAT INTEGRATES WITH THE
DATABASE MANAGERS.

PAYABLES * RECEIVABLES * GENERAL LEDGER*
PAYROLL * SALES ORDER ENTRY *
PURCHASE ORDER ENTRY * INVENTORY *
CHURCH MEMBERSHIP * RENTAL PROPERTY MGT. *
$39.95 each. through March 31st only!

****

ASSISTANCE TO PROGRAMMERS

EXPERT BETA-TESTING
OF ALL PROGRAM FEATURES
FROM USERS POINT OF VIEW.
HINTS FOR IMPROVEMENTS.
*
AID IN PREPARATION OF
PROGRAM DOCUMENTATION :
ORGANIZATION * WRITING
AND PRESENTATION.
*
MARKETING ASSISTANCE
*

CALL OR WRITE

DON'T LOSE SALES
NEEDLESSLY
TEST, DOCUMENT,
AND PRICE YOUR
WORK THE WAY IT
WILL INCREASE
YOUR POTENTIAL

**
+
+ ECS +
RX FOR
+ +
COMPUTERPHOBIACS
+ +
206-857-7878
+ +
+

CUSTOM DATABASE DESIGN

OVER 3 YEARS
EXPERIENCE WITH
THE WORKBASE
SERIES OF POWERFUL
DATABASE SOFTWARE.
*
QUOTES ON
PROTOTYPE
DATABASE TEMPLATES
WITH REPORT AND
CALCULATION
PROCEDURES
*

*
CAN BE DESIGNED
TO RUN IN NEARLY
ANY TYPE OF
BUSINESS

*
WITH WORKBASE
YOU ARE THE
ARCHITECT OF
YOUR OFFICE
MGT. SYSTEM

*

REASONABLE FEES
*

CALL FOR
ANY
DETAILED
DISCUSSION

*

##########

SPECIAL ANNOUNCEMENT !!!

SOON TO BE RELEASED EXCLUSIVELY
by ECS
CALINDEX ver. 3.2
by James Talcott
**
CALL FOR DETAILS ON THIS GREAT
CALENDAR/PROJECT MANAGER
A MUST FOR PROJECT BASED OFFICES

##########

characters until it finds a zero. If this happens try hitting the reset button. This might save you.

Now we will see how to access data that is in a table. A table can consist of numerical or text data. Here we will write a routine that allows us to access a one byte value.

```
START  LDX  #TABLE  ;POINT TO
                    ;TABLE
       LDB  OFFSET  ;GET OFFSET
       ABX          ;ADD B & X
       LDA  ,X      ;GET A BYTE
       STA  RESULT  ;SAVE IT
       SWI
OFFSET FCB  2
TABLE  FCB  Ø
       FCB  1
       FCB  2
       FCB  3
       END
```

In this example, the X index register pointed to the start of TABLE and then the B register was loaded with the offset, which in this case contained the value two. Next the X and B registers were added together with the special instruction ABX which you may remember from part 1Ø of this series. We then load the A register with the number now pointed to by the "effective" address held in the X index register. In the example we would load A with the number two. If the value of the OFFSET were zero, A would have been loaded with zero. If it were one it would be loaded with one from the table. Remember that the zero index begins with the first element pointed to by the index register.

We can also access tables of two bytes using the shift instruction ASL (Arithmetic Shift Left). ASL doubles the value in the offset register allowing us to access data elements that are two bytes long. Here is an example.

```
START  LDX  #TABLE
       LDB  OFFSET
       ASLB         ;DOUBLE
                    ;OFFSET
       LDY  B,X     ;LOAD Y WITH
                    ;2 BYTE DATA
       SWI
OFFSET FCB  1
TABLE  FDB  $ØØØØ
       FDB  $ØØØ1
       FDB  $ØØØ2
       FDB  $ØØØ3
```

In the example the Y register will be loaded with the second table entry $ØØØ1. We can also access multi-dimensional tables or arrays using the MUL or multiply command. In Basic, a two dimension array is written as DIM D(4,5). We access the data in the array by finding the row and column that contain the data. In assembly the formula is:

DIM D(X,Y) = BASE ADDRESS OF ARRAY + NUMBER OF COLUMNS IN ARRAY (Y) * ROW NUMBER + COLUMN NUMBER WE WANT TO LOCATE

To access data in a 4 by 5 array we would do the following.

```
START LDX  #ARRAY ;BASE ADDRESS
      LDB  #5     ;# OF COLUMNS
      LDA  ROW    ;ROW NUMBER
      MUL         ;MULTIPLY
      ADDD COL    ;ADD COLUMN
      LEAX D,X    ;OFFSET INTO
                  ;ARRAY
      LDA  ,X     ;GET DATA
      SWI
ROW   FCB  Ø
COL   FCB  Ø
ARRAY RMB  2Ø     ;RESERVE 2Ø
                  ;BYTES FOR
                  ;ARRAY
      END
```

We use the RMB (Reserve) directive to set aside 2Ø bytes for the array in the program. The Reserve command does not place any values in the memory area reserved. It's up to you to

place the array's data in memory. In the above we would be accessing the Ø,Ø element of the array, to access other data elements in the array all we need do is change the subscripts, just like in Basic. Remember that you can have both numerical and text data in a table. We can use this indexing mode to create a "jump" table using the BSR (Branch to Subroutine) or JSR (Jump to Subroutine). These commands are similar to the GOSUB command in Basic.

```
JUMP TABLE

JUMP   LDX   #TABLE
       LDB   OFFSET
       ASLB
       JSR   B,X
       SWI
OFFSET FCB 1
TABLE  FDB   ROUTØ
       FDB   ROUT1
       FDB   ROUT2
ROUTØ  RTS
ROUT1  RTS
ROUT2  RTS
       END
```

When using a jump table routine like the one above be sure to have a routine for the routine to jump to or you may run into trouble. Check the value in B to see that it doesn't get any larger then you expect. Also be sure to have a RTS (RETURN) instruction in each routine that you jump to so that the program will return correctly. You notice that in the above I simply placed routines at each Routine. Of course you will most likely want to perform some different function at each location. You may also use the JMP (JUMP) in place of the jump to subroutine command if you don't want to return to the calling routine. If you do, be sure to remove the RTS instructions.

Examine these routines and see if you can put them to use in your programs. You may notice that we are beginning to use instruction that we already covered to create our programs. This is how we can build larger Assembly language programs by using the simple commands together to create useful routines. In the next part of this series we will cover some of the remaining instructions and get started at some more advanced Assembly language concepts. See you then.

PRODUCT REVIEWS

This section is open to all producers and dealers of color computer products. We will review your product free of charge and write an editorial on the product. We do not use a rating system but will explain what the product does, and what can be expected from it. Any comments about the review from the firm submitting the product will be printed in a later issue.

DC-4 DISK CONTROLLER

J & M Systems, Ltd. has introduced the DC-4 disk controller. It's features include low price, a DOS selectable switch, and JDOS disk basic. The controller can be purchased with Radio Shack DOS or JDOS. We purchased the JDOS version because we had Radio Shack DOS which we could place in the other socket.

The controller is housed in a metal enclosure with gold plated contacts. We removed the cover and installed the Radio Shack DOS in the socket. After replacing the cover, we plugged in the controller and then plugged a disk drive cable into the output plug on the controller.

Additional Features

JDOS provides some additional features over Radio Shack DOS and we want to give a brief description of some of them.

Automatic line numbering: Just type in AUTO BEGINNING, INCREMENT. Example AUTO 1000,10. After pressing the ENTER key 1000 will be printed and you can add the commands for that line.

LIST: The up and down arrows can be used to scroll up and down through a program. As an example type "LIST 50 ENTER" and line 50 will be listed. To continue listing one line at a time press the down arrow. To list backwards press the up arrow. This is a very useful feature.

ERL: Returns the line number where an error occurred during execution of a basic program. Example A=ERL(0). A is the line number for the error.

ERR: This command returns the error code associated with the error.

ERROR N: This error commands causes an error to go to line number N. Example ERROR 200.

BAUD X: This sets the baud rate. X must be a number from 1 to 7.

RAM: Copies ROM data into RAMS.

RATE: This command changes the head step rate.

The two major features we liked for writing basic programs are the auto line numbering and the up and down arrow statement listings. The ability to change DOS roms with the switch allows using the best of both systems. For example JDOS can be used to write, edit and save basic programs. It also formats disks faster than Radio Shack DOS. Then the computer can be reset and the Radio Shack DOS selected. JDOS is not compatible with some software. An example is COCO MAX. The JDOS manual is very complete and informative. We found the DC-4 controller to be very useful and perform as advertised. It sells for $75 with either Radio Shack or JDOS. For more information contact J & M Systems, Ltd., 15100 Central S.E., Alburquerque, NM 87123.

Color Computer 3- 512K RAMDISKS

We reviewed two ramdisks for the color computer 3 from Cer-Comp and C.R.C Computer Inc. A ramdisk works like a disk drive by allowing programs to be stored within the computer's memory. Files can be accessed from a ramdisk about 20 to 30 times faster than with a disk drive.

Both software ramdisks are on a disk as a machine language program. To activate the ramdisk it is necessary to load and execute the program.

The Cer-Comp program is listed as RAMDISK on the disk. To use it type LOADM "RADISK". It automatically loads, executes the program, and returns to basic with the familar "OK". Drives 2 and 3 are reserved by the program for ramdisks. After executing "RAMDISK" it is necessary to clear the ramdisks unless a disk is to be transferred to one of the ramdisks. The DSKINI command can be used for clearing the ramdisks. Instructions are included for changing the designated ramdisks to drives 1 or 2 although this is not usually needed. A ramtest program is also included for checking the ram.

The C.R.C. Computer programs work in a similar manner. The ramdisk is saved on the disk as "512KUT". After loading and executing the program, the two drives for ramdisks need to be selected. Then the question "INIT?" is asked. For a "Y" response the ramdisks are initialized. Pressing "N" bypasses this and goes to the next section. A printer spooler is included with this software. The program next asks for a "Y" or "N" for using the spooler.

We found both programs to perform as advertised. The C.R.C. program has more features but takes longer to load due to having to answer the prompts asked. If the computer hangs up, it will be necessary to reboot the ramdisk program using software. The data in the ramdisk will be retained most of the time.

The programs are in the $20 price range. For more information contact the following dealers:

Cer-Comp, 5566 Ricochet Ave., Las Vegas, Nevada 89110 (702) 452-0632.

C.R.C. Computers, 10802 Lajeunesse, Montreal, Quebec, Canada H3L 2EB (514) 383-5293.

FKEYS III

FKEYS III is a user programmable function key utility. It can be used on all color computers. For the CC-3 the F1, F2 and CONTROL keys are put to maximum use. For the older computers the down arrow can be used for the control key.

To use the program RUN "MENU". The following options appear:

1- Configure new Fkeys
2- Load custom Fkeys
3- Accept default Fkeys
4- Disable Fkeys
5- DOS modifications

It is easy to configure new Fkeys. To get maximum benefit from the program, several sets of Fkeys could be programmed and saved for particular tasks. There are 20 keys that can be programmed and each key can contain 25 characters in its instructions. To give an example of the type of instructions that can be programed, the following is for the default key function F1:

```
CLS:DIR0:?"FREE:";FREE(0) cr
```

where cr means carriage return.

The DOS modifications allow 35 or 40 tracks, single or double sided, and 6 to 30 ms seek & restore rates. We tried 20 ms instead of the 30 ms we had been using and noticed quite an increase in speed.

We made a few sets of custom Fkeys. The program is easy to follow and the custom keys can be used to great advantage. For example for copying disks to ramdisks and from ramdisks to disks the following can be programmed using F1 and F2:

BACKUP Ø TO 1 Press F1
BACKUP 1 TO Ø Press F2

The carriage return can be programmed also with the key function. The program is easy to use and useful as a programming aid. The cost is $24.95 + $3 S/H. Spectrum Projects, P. O. Box 264, Howard Beach, NY 11414.

NEW PRODUCTS

This section is available free for producers and dealers of color computer products. These products have not been reviewed by us but are included for our reader's information. Dealers we prefer a printed copy and a disk TXT or DAT file to save typing time.

TV BLACKOUT BINGO

This software is compatible with any Color Computer disk system with 32K of memory. TV BLACKOUT BINGO will play up to fifty bingo cards with your favorite TV bingo game. Cards may be entered, sorted, and listed on the user's TV or monitor. The user may check the status of all cards at any time. Side one of the TV BLACKOUT BINGO disk contains software for the original Color Computer, or the CoCo II. Side 2 has special programming for the CoCo 3. All data is compatible between the two programs. The program is not copy protected. However, it is run protected. A code plug is supplied with each program. Users may make backup copies, as they wish.

COLOR BANKBOOK +3

This replaces the original Color Bankbook. COLOR BANKBOOK +3 is compatible with any 32K Color Computer disk system. It also includes a special edition for the CoCo 3. There have been numerious improvements since the original Color Bankbook. COLOR BANKBOOK +3 will be shipped with programming for the original CoCo, or CoCo II, on side one. Special programming for the CoCo 3 is on side 2. All data used with the program is compatible between the two programs. The COLOR BANKBOOK +3 system can completely replace the user's manual check register. It will also print checks, if you have the proper check forms. It maintains a file of regular payees, prints one time only checks, has complete editing, and has provisions for sorting checks, or deposits, for accounting or tax considerations. The COLOR BANKBOOK +3 lists all information on the user's screen or printer. Check register, list of checks outstanding, or lists of account numbers, may be sorted to the screen, or printer. A printer is not necessary to operate the program. COLOR BANKBOOK +3 is not copy protected. However, it is run protected. A code plug is supplied with each program. The owner may make backup copies, for their own use at any time.

The cost of both programs is $19.95 each plus $2.ØØ shipping. Florida residents add $1.ØØ sales tax. Orders may be placed by mail, or by calling 1-8ØØ-628-2828, extension 552. Sunrise Software, 89Ø1 NW 26 St., Sunrise, FL 33322.

INTRODUCING DYPRINT

BANNER

Now you can print **LARGE** signs for special occassions such as birthdays, parties, or yard sales. Even make your own **FOR SALE** signs when you need to sell that old car or lawnmower. Banner uses standard print characters and is compatible with any printer. The characters are formed by a 21 x 27 dot pattern and are printed sideways across the paper. The basic character can be expanded up to 4 times for making large characters up to a full page.

The printer parameters can be used to expand the size and quality of the signs. For example high density signs can be printed with printers that use compressed characters. Darker signs can be printed by using double strike.

MAXPRINT

MAXPRINT allows graphics to be blown up and printed on a standard printer. Any PMODE 4 picture generated by COCOMAX, MAGIGRAPH, VIDEO DIGITIZERS, or BASIC can be printed. This allows a large picture or poster to be made. The program supports all 8 graphics pages for a total of 12288 bytes. **MAXPRINT** prints 8 characters per byte for a total of 98304 characters.

The graphics picture is 256 characters wide and is printed with 2 passes for the 128 character per line mode or 8 passes for the 32 character per line mode using large characters. The results from each pass can be trimmed and taped together to form a large blown up picture.

Use **MAXPRINT** to blow up pictures of friends and family and make posters announcing sales or special events.

The **DYPRINT** package contains both **BANNER** and **MAXPRINT**. The cost is only **$19.95** plus $3 shipping for tape or disk.

DEALER INQUIRIES INVITED

Checks, VISA or MC Cards

DYNAMIC ELECTRONICS Inc.
Box 896 (205) 773-2758
Hartselle, AL 35640

MEMORY SAVER 2

MEMORY SAVER 2 is a memory backup system that prevents programs and data from being lost due to power failures. It was especially designed to protect the 512K memory of the new Color Computer 3 but can be used for all color computers. The assembly consists of a 6 volt, 1.2 ampere hour sealed rechargeable battery, an electronic control circuit, and a small switch. The small battery and assembly mount under the keyboard. The switch enables or disables the MEMORY SAVER 2. It should be disabled when the computer is turned off to prevent the battery from discharging. The battery charges from the computer and automatically supplies power to the memory chips when power fails.

Installation involves one solder connection and drilling a small hole for the switch. Eyelets slide over two pins of one of the memory chips completing the installation. The MEMORY SAVER 2 will power a color computer 3 from 30 minutes to an hour depending upon its current requirements. The MEMORY SAVER 2 costs $39.95 +$3 shipping. Dynamic Electronics Inc., Box 896, Hartselle, AL 35640 (205) 773-2758.

OPERATING HINT

The vector in locations 135-6 determine the screen position of the cursor. The cursor can be moved to any screen location by poking values into these locations. This works with the CoCo 3 in the 32 character display mode,

INTERFACING COMPUTERS

MEASURING

TEMPERATURE

INTRODUCTION

For the past few months we have been looking at methods of utilizing the joystick ports. The joystick ports are connected to an analog to digital (A/D) converter that converts a voltage from Ø to 5 volts to a digital number from Ø to 63. As a joystick is moved, the center arm of a potentiometer is moved and gives an output voltage that is an indication of the relative position of the joystick. This voltage is converted into a number by the JOYSTICK command. There are 4 joystick ports in a color computer. A joystick contains two potentiometers and plugs into a jack on the computer. One potentiometer is moved with up and down motion and the other is moved with right and left motion.

A voltage of 5 volts is present within a joystick plug. We have been looking at connecting two resistors in series across this 5 volts and ground. The junction of the two voltages goes to the joystick input. For measuring voltage, we can connect the unknown voltage to the joystick input. We showed how to connect a series resistor in order to measure a voltage greater than 5 volts. Also we showed how to use a diode to change alternating current(ac) voltage to direct current(dc) and make ac voltage measurements.

Last month we returned to the two resistor configuration and showed how to determine the value of an unknown resistor. We are showing how to use passive elements for making measurements. A passive element is one that does not provide amplification but gives an output that is related to the input. For example a resistor is a passive component. It obeys OHM'S law. On the other hand, a transistor is an active component. It can be operated in many modes depending upon biasing. A resistor has only one mode. The voltage across a resistor is always equal to the resistance value multiplied by the current flowing through it by Ohm's law.

Let's state what we are doing in another way. We are limiting ourselves to the 5 volts that appears on the joystick ports. We are not using any amplifiers because additional voltages are usually required. We are sacrificing linearity and sensitivity in exchange for simplicity. Perhaps when we finish our present objective, we could add refinements. For example we can measure voltages from Ø to 5 volts. What about negative voltages and very small voltages. These limitations can be overcome with more sophisticated circuits.

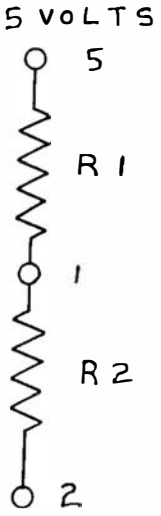


FIGURE 1

MEASURING TEMPERATURE

Refer to Figure 1. From Ohm's law the current is

(1) I=5/(R1+R2)

The voltage at the center of the two resistors is

(2) V = I*R2=5*R2/(R1+R2) = 5/(1+R1/R2)

The last expression is obtained from algebra. Notice that if R1=Ø then V=5. As R1 gets larger than the voltage at the junction of the resistors becomes smaller and smaller.

Last month we looked at varying one of the resistors and obtaining the value of the resistor by operating on the joystick voltage reading. We can use the same technique for calculating the temperature.

THERMISTORS

One way to measure temperature is to use a resistor that changes value with temperature which is called a thermistor. There are two kinds of thermistors. The first kind has a positive temperature of coefficient. This means that as the temperature increases, the resistance increases. The second

kind of thermistor has a negative coefficient of resistance. For this type the resistance goes down as the temperature increases.

A graph of the resistance versus temperature of a negative coefficient of resistance thermistor is shown in Figure 2. Referring back to Figure 1. Notice that if either of the resistors is a thermistor then the joystick input on pin 1 will vary as the temperature varies. The problem is to get the proper reading for each temperature.

We looked at several methods of converting the response of a thermistor so we could use it with the computer. First we tried to write an equation for the response in Figure 2. This did not work out very good over the total range. Next we looked at straight line approximations. The following data points were used to write 3 line segments.

	R	T
(1)	4K	1Ø2
(2)	1Ø.5K	77
(3)	25K	44
(4)	55K	16

The equation for a line through points (1) and (2) is

(a) T=-3.846E-3 * R + 117.38.

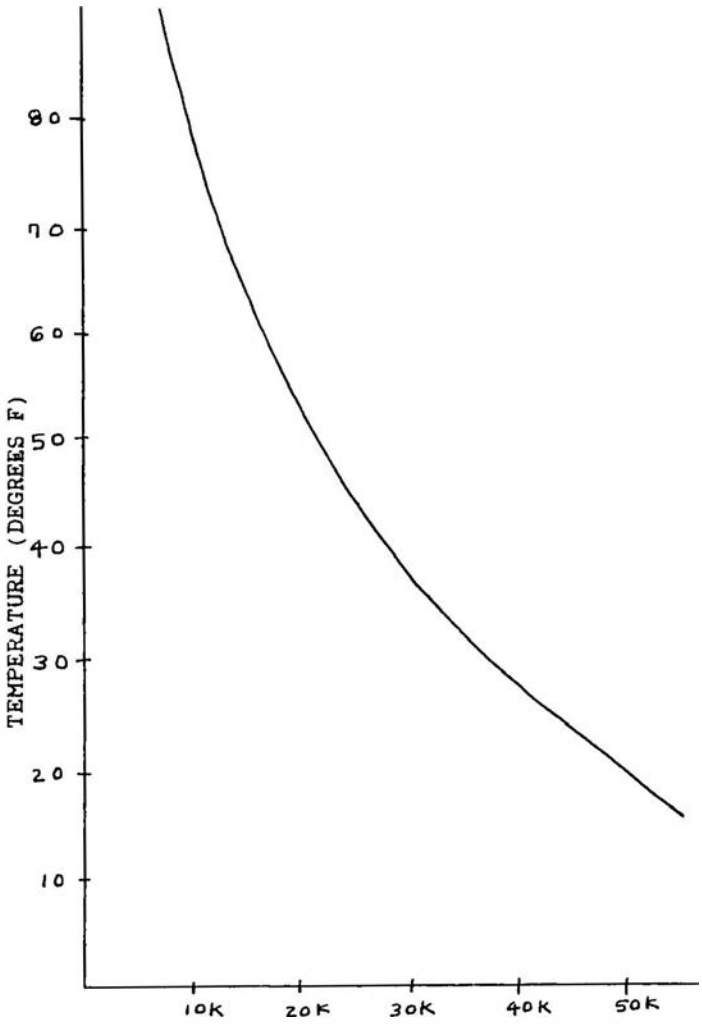
The equation for a line through points (2) and (3) is

(b) T=-2.276E-3 * R + 1ØØ.897.

The equation for a line through points (3) and (4) is

(c) T=-9.333E-4 * R + 67.333.

Now that we have equations that approximate the temperature, we need an equation that converts the joystick voltage to a resistance. Again refer to Figure 1. We will let R2 be the



THERMISTOR RESPONSE CURVE

FIGURE 2

termistor and use the symbol R to represent it. Using circuit analysis theory we can derrive the following equation:

(d) R=R1/(63/JS-1) where JS is the joystick value.

Remember the JS reading represents a percentage of 5 volts. The value of R from equation (d) can be used to calculate the temperature in equations a,b, or c.

We wrote a program to test the equations against the measured thermistor response. The first part of the program asks for the value of R1. We looked at values of 1ØØØØ, 15ØØØ, and 27ØØØ. The best value seemed to

be 10000. The program then asks for a joystick value from 0 to 63. It then gives the calculated temperature.

Next month we want to continue with our digital thermometer and give construction details. Two thermometers can be built into one joystick plug. One could be used for measuring outside temperature and the other for inside temperature. Also we can use the temperature measurement for controlling an air conditioner or heater.

TEMPERATURE DEMO PROGRAM

```
10 'TEMPERATURE DEMO PROGRAM
20 'cOPYRIGHT (c) 1987
30 'dYNAMIC eLECTRONICS iNC.
40 '
85 INPUT"ENTER RESISTOR";R1
100 INPUT"ENTER JOYSTICK VALUE
    0-63";JS
110 R=R1/(63/JS-1)
115 IF R<10500 THEN GOTO 200
120 IF R>25000 THEN GOTO 250
130 GOTO 220
190 '
200 T=-3.846E-3*R+117.38
210 PRINT USING "###.#";T:PRINT:
    GOTO100
220 T=-2.276E-3*R+100.897:
    GOTO210
250 T=-9.333E-4*R+67.333:
    GOTO 210
```

MULTI-PAK DISK RESET

When using a disk drive with a multi-pak interface, you can sometimes reset the computer when the rear reset button does not work. To do this move the selector switch to a position that is not used and then press the reset button. The computer will reset and come up with an extended basic message. POKE 113,0 and then move the switch back to the slot containing the disk controller. Then again press the reset button and the normal disk basic message should appear. Infromation is still retained within the computer. A basic program can be restored using our RESTORE page -1 program

---

HAM RADIO & COMPUTERS

by

Bill Chapple W4GQC

MORSE CODE BY COMPUTER

Last month the FCC enhanced the Novice privileges. They were given voice privileges on 28 MHz and 220 MHz plus some higher frequencies. This means that they can use single sideband (SSB) on 28 MHz and frequency modulation (FM) on 220 MHz. Also with a Novice class license pictures can be exchanged between hams using slow scan television. This is a subject we want to cover in the future. Also teletype, packet, and repeater operation is available under the new regulations.

The novice class examination is the easiest to get requiring the ability to copy 5 words a minute of Morse code and answer a short written test. This is encouraging to me as I am sure these increased privileges will enhance the growth of amateur radio.

Last month we gave a tone decoder circuit that gives a voltage out as long as a tone is present. This circuit can be used for many applications. Our main purpose for the tone detector was to allow the computer to copy and print the characters generated by Morse code. This month I was able to achieve our goal although more work is needed.

I set up a test using a color computer 3 to generate the Morse code. The audio of the CC-3 was amplified and applied to the tone decoder of our interface circuit. The interface circuit was connected to the RS-232 port of an older color computer. With the CC-3 automatically sending Morse code, the value at 65314 of the older computer changed with the characters indicating that the tone decoder was working and information was being received.

SOFTWARE

The hardware was completed and software was needed. How can we write software to decode Morse code? First let's look at automatic speed tracking.

Let's let a dot be the basic time element. Then we can define the following assuming that S represents the time for a dot and A represents the time for an element:

(1) IF A<S/2 THEN S=A. This adjusts the reference for a faster speed.

(2) IF A>4*S THEN S=A. This adjusts the reference for a slower speed. A dash is 3 times the length of a dot. If the element was longer than 4 times a dot then the speed needs to be changed.

(3) IF A<=S THEN W=Ø. We will let W represent the element. If A=S then the element must be a dot and W=Ø. W can be less than S but not less than 2*S or a speed change will occur and a new S defined as in (1).

(4) IF A>=2*S THEN W=1. If the time is longer than 2 times a dot then the element is a dash.

ACCUMULATING ELEMENTS

The preceeding routines allow us to determine if an element is a "Ø" or "1". Now we need a method of combining the elements to form the character. We used the variable Q to represent the result or the detected Morse character. We first let Q be one and then multiply it by 2. This shifts it left one position. We then add the next element and shift it left by multiplying by 2. This continues until we finish the character. The following statement does this for us:

```
Q=2*Q+W:IF Q>128 THEN Q=Ø
```

OFF TIMING

We let the variable B accumulate time when there is no input. This will be between elements and to determine if a character was sent and a word space. We used the following for determing this:

```
(1) IF B=2*S THEN ?A$(Q):Q=1
    'END OF CHARACTER ?CHARACTER

(2) IF B=5*S THEN ?" ";:Q=1
    'PRINT WORD SPACE
```

CHARACTER ARRAY

Notice in (1) that we used the command ?A$(Q). We set up an array to contain the characters. Let's look at a couple of examples to see how the array was set up. An "A" is a dot-dash. We let Q be 1 as we start accumulating the bits and shifted this 1 left by multiplying it by 2. The easy way to set up the array is the write the bits for the character and add a 1 in the position next to the left element. Therefore we will have 1Ø1 for "A" since a dot-dash is a Ø1. So the "A" will be 1Ø1 or 5. A "B" is 1ØØØ and with the extra 1 to the left a "B" becomes 11ØØØ or 24.

After we determine Q then we can look up A$(Q) from the array and print the character. The array is setup using the DIM command. We setup for 13Ø elements although all of these are not needed. We just defined the ones that we are using in the definitions.

The program worked in the double speed mode up to about 15 words a minute. This needs to be improved and may require a machine language subroutine or two to speed up the timing process. We will improve on this next month.

MORSE COPIER

This program requires the interface circuit presented in the past few issues plus the tone decoder present last month. Additional information and improvements will be given next month.

```
2 'MORSE CODE COPIER
4 'COPYRIGHT (c) 1987
6 'DYNAMIC ELECTRONICS INC.
8 '
1Ø GOSUB 24Ø
15 INPUT"ENTER 1 FOR HIGH SPEED
   ";HS: IF HS<>1 THEN 3Ø
2Ø POKE65495,Ø
3Ø Q=1:A=Ø:B=Ø:C=Ø:S=1
4Ø X=PEEK(65314) AND 1
5Ø IF X>Ø THEN B=Ø:A=A+1
6Ø IF X=Ø THEN 9Ø
7Ø GOTO4Ø
8Ø '
9Ø IF B>Ø THEN 19Ø 'END OF ON TI
   ME
1ØØ FOR J=1 TO 1:NEXT
11Ø IF A<S/2 THEN S=A' COMPENSATE
   FOR FASTER SPEED
12Ø IF A>4*S THEN S=A 'SLOWER SPE
   ED
13Ø IF A<=S THEN W=Ø 'CHAR =DIT
14Ø IF A>=2*S THEN W=1 'CHAR =DA
   SH
15Ø Q=2*Q+W:IF Q>128 THEN Q=Ø
16Ø B=B+1:A=Ø
17Ø GOTO 4Ø
18Ø '
19Ø 'IF B=1 THEN PRINT"Q="Q;
2ØØ FOR J=1 TO 1:NEXT
21Ø B=B+1:IF B=2*S THEN PRINTA$(Q
   );:Q=1
22Ø IF B=5*S THEN PRINT" ";:Q=1
```

```
230 GOTO 40
240 DIM A$(130)
250 A$(5)="A":A$(24)="B"
260 A$(26)="C":A$(12)="D"
270 A$(2)="E":A$(18)="F
280 A$(14)="G":A$(16)="H"
290 A$(4)="I":A$(23)="J
300 A$(13)="K":A$(20)="L
310 A$(7)="M":A$(6)="N
320 A$(15)="O":A$(22)="P
330 A$(29)="Q":A$(10)="R
340 A$(8)="S":A$(3)="T
350 A$(9)="U":A$(17)="V
360 A$(11)="W":A$(25)="X
370 A$(27)="Y":A$(28)="Z
380 A$(63)="0":A$(47)="1
390 A$(39)="2":A$(35)="3
400 A$(33)="4":A$(32)="5
410 A$(48)="6":A$(56)="7
420 A$(60)="8":A$(62)="9
430 A$(85)=".":A$(115)=",
440 A$(76)="?":A$(97)="*
450 A$(0)=CHR$(8) 'ERROR
    GENERATES BACK SPACE
460 RETURN
470 END
```

RS-232 BAUD RATES

By **poking appropriate** values into **memory** locations 149 and 150 the **RS-232** port can handle a variety of baud rates.

149	150	Rate
4	88	50
2	227	75
1	246	110
1	153	134.5
1	110	150
0	180	300
0	87	600
0	40	1200
0	25	1800
0	23	2000
0	18	2400
0	10	3600
0	7	4800
0	3	7200
0	1	9600

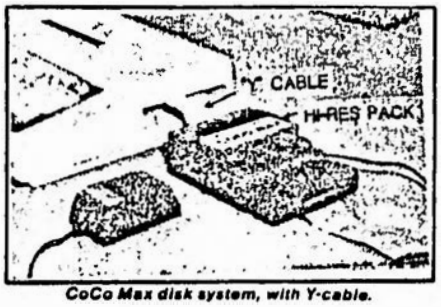
Use double speed poke for 19200 baud.

CoCo Max™ II

You'll use it all the time and love using it.

What is CoCo Max?

Simply the most incredible graphic and text creation "system" you have ever seen. A Hi-Res Input Pack (more on the pack later) is combined with high speed machine language software. The result will dazzle you.



CoCo Max disk system, with Y-cable.

Is CoCo Max for you?

Anyone who has ever held a pencil or a crayon for fun, school or business will love it. A 4 year-old will have fun doodling, a 15 year-old will do class projects and adults will play with it for hours before starting useful applications (illustrations, cards, artwork, business graphics, flyers, charts, memos, etc.) This is one of the rare packages that will be enjoyed by the whole family.

What made CoCo Max an instant success?

First there's nothing to learn, no syntax to worry about. Even a child who can't read will enjoy CoCo Max. Its power can be unleashed by simply ***pointing*** and ***clicking*** with your mouse or joystick. With ***icons*** and ***pull down menus,*** you control CoCo Max intuitively; it works the same way you think.

Don't be misled by this apparent simplicity. CoCo Max has more power than you thought possible. Its blinding speed will astound you.

It lets you work on an area 3.5 times the size of the window on the screen. It's so friendly that you will easily recover from mistakes: The ***undo*** feature lets you revert to your image prior to the mistake. As usual, it only takes a single click.

Later, we will tell you about the "typesetting" capabilities of CoCo Max II, but first let's glance at a few of its graphic creation tools:

With the ***pencil*** you can draw free hand lines, then use the ***eraser*** to make corrections or changes. For straight lines, the convenient ***rubber-banding*** lets you preview your lines before they are fixed on your picture. It's fun and accurate. Lines can be of any width and made of any color or texture.

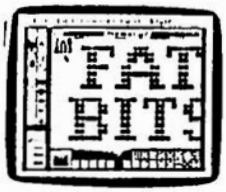
The ***paint brush,*** with its 32 selectable brush shapes, will adapt to any job, and make complicated graphics or calligraphy simple.

For special effects, the ***spray can*** is really fun: 86 standard colors and textures, all available at a click. It's like the real thing except the paint doesn't drip.

CoCo Max will instantly create many shapes: circles, squares, rectangles (with or without rounded corners), ellipses, etc. Shapes can be filled with any pattern. You can also add hundreds of custom patterns to the 86 which are included.

The ***Glyphics*** are 58 small drawings (symbols, faces, etc.) that can be used as rubber stamps. They're really great for enhancing your work without effort.

Pull down menus



Zoom in!

Control Over Your Work

CoCo Max's advanced "tools" let you take any part of the screen, (text or picture) and perform many feats:

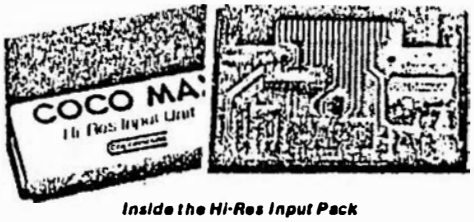
• You can move it around • Copy it • Shrink or enlarge it in both directions • Save it on the electronic ***Clipbook*** • Flip it vertically or horizontally • Rotate it • Invert it • Clear it, etc. etc.

All this is done instantly, and you can always ***undo*** it if you don't like the results.

For detail work, the ***fat bits*** (zoom) feature is great, giving you easy control over each pixel.

To top it all, CoCo Max II works in color. Imagine the pictures in this ad in color. If you own a Radio Shack CGP-220 or CGP-115, you can even print your work in full color!

There is so much more to say, such as the capability to use CoCo Max images with your BASIC programs, the possibility to use CoCo Max's magic on any standard binary image file. There are also many advanced features such as the incredible ***lasso.***



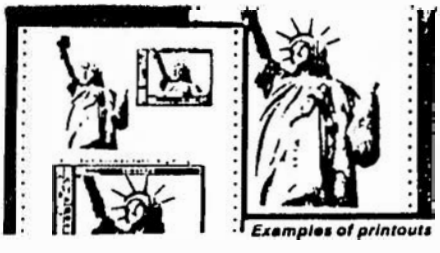
Inside the Hi-Res Input Pack

Why a Hi-Res Input Pack?

Did you know that the CoCo joystick input port can only access 4096 positions (64x64)? That's less than 10% of the Hi-Res screen, which has 49152 points! (256x192). You lose 90% of the potential. The Hi-Res Input Pack distinguishes each of the 49152 distinct joystick or mouse positions. That's the key to CoCo Max's power. The pack plugs into the rom slot (like a rom cartridge). Inside the pack is a high speed multichannel analog to digital converter. Your existing joystick or mouse simply plugs into the back of the Hi-Res Pack.

Electronic Typesetting...

You'll be impressed with CoCo Max's capability. Text can be added and moved around anywhere on the picture. (You can also rotate, invert and flip it...) At a click, you can choose from 14 built in ***fonts*** each with 16 variations. That's over 200 typestyles!



Examples of printouts

Printing Your Creations

There are a dozen ways to print your work. All are available with a click of your joystick (or mouse) without exiting CoCo Max. Your CoCo Max disk includes drivers for over 30 printers!

All the CoCo Max pictures are **unretouched** screen shots or printouts (Epson RX-80).

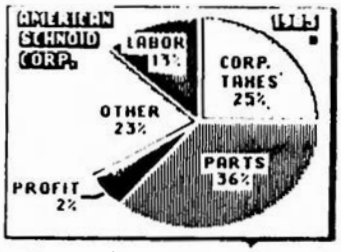
CoCo Max II™

The whole family will enjoy CoCo Max. Here are a few examples of the possibilities.

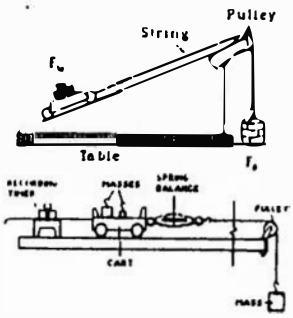
All these pictures are **unretouched** screen photos or printouts (on an Epson RX-80).

❶ *Publish a newsletter or bulletin*

❷ *Fun for children while stimulating creativity.*

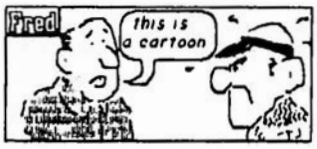


❸ *Business graphs, charts, diagrams. Also memos*



❹ *Junior's homework and science projects. Term papers too!*

❼ *Video portrait (with optional digitizer).*

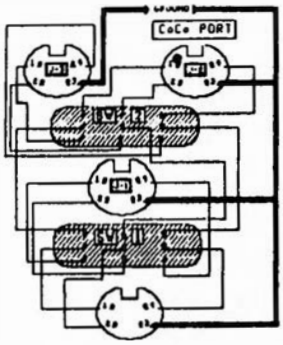


❽ *This is a cartoon.*

CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max
CoCo Max

❺ *Over 200 typestyles to choose from! generate flyers.*

❻ *A new way to express your imagination.*



❾ *schematics and floor plans.*

CoCo Max II
CoCo Max II
CoCo Max II
CoCo Max II

❿ *Logos and letterheads.*

System Requirements:

Any 64K CoCo and a standard joystick or mouse. (The koala pad and the track ball work, but are not recommended.)
Disk systems need a Multi-Pak or our Y-Cable. CoCo Max is compatible with any Radio Shack DOS and ADOS.
Note: the tape version of CoCo Max includes almost all the features of CoCo Max II except *Shrink, Stretch, Rotate,* and *Glyphics.* Also, it has 5 fonts instead of 14.
CoCo Max is not compatible with JDOS, DoubleDOS, MDOS, OS-9, the X-pad, and Daisy Wheel Printers.

Printers Supported:

Epson MX, RX, FX and LX series, Gemini, Star, Micronix, Delta 10, 10X, 15, 15X, SG-10, Okidata 82A, 92, 93, C. Itoh Pro-writer, Apple Image-writer, Hewlett-Packard Thinkjet, Radio Shack DMP 100, 105, 110, 120, 200, 400, 500, Line Printer 7, Line Printer 8, TRP-100, CGP-220. (DMP-130 use Line Printer 8), PMC printers, Gorilla Banana.
Color printing: CGP-200, CGP-115

Pricing

CoCo Max on tape $69.95
with Hi-Res Pack and manual
CoCo Max II (disk only) $79.95
with Hi-Res Pack and manual.
Upgrade: CoCo Max to CoCo Max II
New disk and manual $19.95
New features of CoCo Max II: 14 fonts and glyphic font, dynamic shrink and stretch, rotate, multiple drive capability, 68 page scrapbook, point and click file load, color printer drivers, full error reporting.
Upgrade: CoCo Max tape to disk
manuals, disk and binder $24.95
Y-Cable: *Special Price* $19.95
Super Picture Disks #1, #2, and #3
each: $14.95
All three picture disks $29.95

Checks, VISA & MC
Add $3 Shipping

Font Editor Option

A font is a set of characters of a particular style. CoCo Max includes 15 fonts. You can create new fonts of letters, or even symbols or graphics with the font editor. Examples: set of symbols for electronics, foreign alphabets, etc. $19.95

Video Digitizer DS-69

This new Low Cost Digitizer is the next step in sophistication for your CoCo Max system. With the DS-69 you will be able to digitize and bring into CoCo Max a frame from any video source: VCR, tuner, or video camera. Comes complete with detailed manual and C-SEE software on disk. Multi-Pak is required.
New Low Price Save $50 ..... $99.95
New: faster DS-69A $149.95

COLORWARE

DYNAMIC ELECTRONICS Inc.
Box 895 (205) 773-2758
Decatur, AL 35602

COLOR COMPUTER 3

In this series we have been looking at the features of the color computer 3 and comparing its features with the other computers. This month we upgraded our CC-3 to 512K and reviewed a couple of ramdisk programs. A memory of 512K allows two ramdisks to be configured. This is the same memory the Banker RAM provided for other computers. In fact the performance is similar.

After configuring for the two ramdisks, we copied a disk into one of the ramdisks and then made a backup copy of the disk. This worked good and does not require any disk swaps. If you have ever backed up a disk using the disk swap method, you will really appreciate using the ramdisk. The backup time was much less than it would be using two disk drives.

ERROR TRAPPING

This month we want to look at some error trapping commands. These are new commands and are not available on the other models. Some disk operating systems contain error trapping which can be used on the CC-2. Previously when an error occurred the program stopped and an error message was printed on the screen. With error trapping, the program can be forced to go to specified line numbers when the error occurs.

ON ERR GOTO X

This command tells the computer the line number to go to when an error occurs. A good example is a divide by Ø. If this happens the computer will print a /Ø error message and stop. The line after the GOTO can be used to print instructions on the screen or branch to another location in the program. Let's look at an example program.

```
 5 ON ERR GOTO 5Ø
 1Ø INPUT"ENTER NUMERATOR";X
 2Ø INPUT"ENTER DENOMINATOR;:Y
 3Ø Q=X/Y:?"QUOTIENT ="Q
 4Ø GOTO 1Ø
 5Ø ?"DENOMINATOR MUST BE GREAT
    ER THAN Ø TRY AGAIN";:GOTO
    2Ø
```

ERNO

This command returns the error number after branching to the line number indicated by the ON ERR GO TO command. A detail of the error can then be printed. In fact a subroutine could be included with the program to print comments on any error.

There are 4Ø error code numbers and the computer prints a 2 letter code when an error occurs. For error Ø the code is NF and this means Next Without For.

The error numbers, codes, and comments are listed on page 321 of the programming manual. At the end of this editorial is a program that describes errors. It can be merged with another program.

ERLIN

This command returns the line number that contains the error. It can be used to cause a branch depending upon the line number causing the error. One way to do this is as follows:

IF ERLIN=X THEN Y where X and Y are line numbers.

ON BRK GOTO X

This command allows the operator to press the break key and cause the program to branch to line X.

ERROR TRAPPING PROGRAM

The following program will print a description of error when one occurs. It can be merged with other programs if saved in ASCII format. The program can be used with the CC-3 and other computers that have a DOS with error trapping.

```
1 'ERROR TRAPPING PROGRAM
2 'COPYRIGHT (c) 1987
3 'dYNAMIC eLECTRONICS iNC.
4 'MERGE WITH OTHER PROGRAMS
5 ON ERR GOTO 49000
48999 '
49000 XJ=ERNO:IF XJ=38 THEN GOSU
   B 50380 ELSE IFXJ=39 THEN GOS
   UB 50390:GOTO 55000
49010 XJ=XJ+1:IF XJ>13 THEN 4903
   0
49020 ON XJ GOSUB 50000,50010,50
   020,50030,50040,50050,50060,5
   0070,50080,50090,50100,50110,
   50120:GOTO 55000
49030 XK=XJ-13: ON XK GOSUB 5013
   0,50140,50150,50160,50170,501
   80,50190,50200,50210,50220,50
   230,50240:GOTO 50000
50000 PRINT"NEXT WITHOUT FOR";:R
   ETURN
50010 PRINT"SYNTAX";:RETURN
50020 PRINT"RETURN WITHOUT GOSUB
   ";:RETURN
50030 PRINT"OUT OF DATA";:RETURN
50040 PRINT"ILLEGAL FUNCTION CAL
   L";:RETURN
50050 PRINT"OVERFLOW";:RETURN
50060 PRINT"OUT OF MEMORY";:RETU
   RN
50070 PRINT"UNDEFINED LINE";:RET
   URN
50080 PRINT"BAD SUBSCRIPT";:RETU
   RN
50090 PRINT"ATTEMPT TO REDIMENSI
   ON ARRAY";:RETURN
50100 PRINT"DIVISION BY ZERO";:R
   ETURN
50110 PRINT"ILLEGAL DIRECT STATE
   MENT";:RETURN
50120 PRINT"TYPE MISMATCH";:RETU
   RN
```

```
50130 PRINT"OUT OF STRING SPACE"
   ;:RETURN
50140 PRINT"STRING TOO LONG";:RE
   TURN
50150 PRINT"STRING FORMULA TOO C
   OMPLEX";:RETURN
50160 PRINT"CANNOT CONTINUE";:RE
   TURN
50170 PRINT"BAD FILE DATA";:RETU
   RN
50180 PRINT"ALREADY OPEN";:RETUR
   N
50190 PRINT"DEVICE NUMBER ERROR"
   ;:RETURN
50200 PRINT"INPUT/OUTPUT ERROR";
   :RETURN
50210 PRINT"BAD FILE MODE";:RETU
   RN
50220 PRINT"FILE NOT OPEN";:RETU
   RN
50230 '
50240 PRINT"DIRECT STATEMENT";:R
   ETURN
50380 PRINT"HIRES GRAPHICS ERROR
   ";:RETURN
50390 PRINT"HIRES PRINT ERROR;":
   RETURN
50400 '
55000 PRINT" ERROR IN LINE";ERLI
   N:END
```

BASIC PROGRAMMING

In this series we have been showing how to write basic programs. There is much interest in writing programs that will handle files longer than 32K. In our February issue, we gave a file handling program that allowed creating an address file that contained 100 addresses. A sort routine was included to allow the file to be sorted on names, addresses, or telephone numbers.

MULTIPLE FILES

This month we want to look at handling many files. The address file we developed occupied memory from 9999 to 19999. The value in 9999 contained the number of files used. This number can be any value from 0 to 99. We reserved memory from 20000 to 29999 so that we could load two files and sort both of them together.

Why would we want to sort the files? Suppose you were making a telephone directory. We would want names to be listed alphabetically. For bulk mailing the addresses have to be sorted by zip codes. This is what we need when we go to mail Dynamic Color News.

Let's reserve 15 files on a disk for our addresses. We will use the following names for the files:

X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,
X11,X12,X13,X14,X15

SORT PROCEDURE

With our program we will want to sort all files so that X1 contains the smallest and X15 contains the largest. This month we want to go through a procedure that we will formulate into a program. Let's concentrate on X1 and X2. First we will load X1 into the computer and move it to upper memory. Next we will load X2. Remember the files load at 10000. Now let's compress the two files. In other words we will fill in the spaces occupied by blank files in X2 if there are any. As an example if the lower 10000-19999 memory has 67 addresses then we want to move the second file so that its

first address is file number 67. Remember Ø is the fist file. After compressing the files we will have one file of up to 2ØØ addresses. Now we will sort the 2ØØ address file.

After sorting, the lower addresses will be in the lower file. Let's exchange the files and save the lower file as X2. X1 will be in upper memory and will contain the lower addresses.

Next we will load in X3 and repeat the procedure. We will continue with all files. After finishing X15, the file in the computer will contain the lowest addresses. We will save it as X1.

The procedure will repeat comparing X2 with the rest, and saving a new X2 which will contain the next lowest addresses. Next we will compare X3 with the rest and continuing.

When we have finished, the files will be sorted with X1 containing the smallest and X15 containing the largest. This will handle 15ØØ names.

Let's write some steps in accomplishing the sort. First of all we will need some memory locations for housekeeping. Let's reserve the following:

```
9999 Number of files from
     10000-19999
9998 Number of files from
     20000-29999
9997 disk file #
```

The following is an example of the sort routine. Notice that the procedure for sorting files is the same as sorting the individual addresses:

```
1 FOR K=1 TO XZ-1
2 LOADM XK
3 FOR J=2 TO XZ
4 LOADM XJ
5 EXCHANGE XJ-1 AND XJ
6 COMPRESS FILES
7 SORT FILES
8 EXCHANGE FILES-PUT SMALLEST
  IN UPPER MEMORY
9 SAVE LOWER AS XJ
1Ø NEXT J
11 EXCHANGE FILES
12 SAVEM XK:NEXT K
```

Now let's consider the things we will have to do. Loading and saving files are standard procedure with which we are familiar. Compressing files will involve moving the upper file down in memory to fill in the holes on the lower file. This will involve moving blocks of data and changing the upper and lower numbers of files in 9999 and 9998.

Exchanging files will involve switching the lower files with the upper files. We have developed block moving machine language subroutines which we can use for this purpose.

We will need to be able to sort files. This has already been developed and we can use the sort routine in our February issue for this purpose.

SUMMARY

A color computer generally has 32K that can be used. We developed an address file that allows up to 1ØØ addresses to be placed into memory from 1ØØØØ to 19999. We can move addresses up to 2ØØØØ-29999 and contain 2 of these 1ØK files within our computer. This will allow us to sort the two files.

We will allow up to 15 of these files to be placed on a disk with designations X1 through X15. Our program will pull these files, operate on them, and construct a new set of files X1-X15 so that X1 contains the smallest addresses and X15 contains the largest addresses.

The program will work for a ramdisk or regular disk drive. With a ramdisk the files will be able to be loaded or saved much faster than with a disk.

Next month we will continue and present details on the various parts of the program.

EDITOR'S COMMENTS

Our color computer 3 is working well. It has taken a little adjustment to get used to its special features. This month we upgraded it to 512K and reviewed two ramdisk software programs. A floppy disk will hold about 180K of data and two disks require about 360K. Therefore 512K can contain the contents of two disks plus leave extra memory for the computer. I made some copies of one of our disks using the ramdisk. It worked very well and is much faster than using two disk drives. I could not tell any difference in the performance over the Thunder ram or Banker ram used on the older computers.

We designed a battery backup for the color computer 3. This worked very well and will mount inside the case. There is a potential heat problem with the 512K memory. We want to look at this problem and perhaps suggest a solution next month. For the older computers we solved the heat problem by providing an outside heat sink and regulator assembly. I think we can do this for the color computer 3. With the battery backup and heat reducer assembly the color computer 3 can be left on all of the time. With two ramdisks loaded with programs, this would be faster than even a hard disk. I want to experiment with this and will of course give you information as it is completed.

In our interfacing computers section we are looking at measuring temperature with the computer. This will allow us to make a temperature controller. The cassette relay could drive a larger relay that could control a heater or air conditioner. The temperature sensor will give information back into the computer so the heating or cooling source can be turned on or off as needed. Two sensors can be built into one joystick connector and could be used to measure inside and outside temperature.

John Galus is continuing with our Machine Language Programming series. He is doing a very good job of explaining how to write Assembly Language programs. If you are interested in being able to write machine language subroutines or programs then you will want to study his material.

In our Ham Radio series we present this month a Morse Code Copier. This worked well on the bench using two color computers. We had a speed limitation problem which we will resolve next month. There is much material we want to cover in this series.

We are continuing our EPROM series. We have put basic programs into cartridges and want to show how to do this.

I want to thank each of you who recently renewed or signed up for a new subscription. We want to keep our subscription price down so Dynamic Color News can be affordable to everyone who wants a copy. We had to increase our subscription rate to Canada because of the extra postage required.

We need more letters. The questions we have been asked help us know what information you need.

QUESTIONS & ANSWERS

These are questions that have been asked us. If you have a question that you would like for us to answer send it to us at Box 896, Hartselle, AL 35640. We will print our answers here. For a personal reply send $10 with your question.

QUESTION: I have an old color 64K computer and am considering upgrading to the color computer 3. Do you recommend this?

QNSWER: Upgrading will depend upon your requirements. A 64K computer will run most programs. In fact we don't know of any it will not run except those especially for the color computer 3. Unless you want the additional features we would not recommend changing. If you had an old 4K computer then it might be worthwhile purchasing a 64K color computer 2 rather than trying to upgrade it.

CLASSIFIED ADS

1. 10 cents a word, $3 minimum.
2. Name, Address, & Telephone listed free.
3. Send payment with ad.
4. Closing date 1st of the preceeding month. Ex. Nov ad closing is Oct. 1.
5. No X-Rated ads.

We have very, very LOW PRICES on paper, ribbons, disks, and other supplies. Send $1 for our FULL-COLOR catalog and receive a $2 credit. Marty's, 44 S. Front, Saint Clair, PA 17970

Premium Quality Disks. You don't have to pay a lot for QUALITY disks. Our disks are boxed in tens complete with labels, sleeves, and write protect tabs. Don't confuse these with cheaper disks as they carry a lifetime waranty and will be replaced should they become defective. DSK-1 SSDD for CoCo $6.95 /box, DSK-2 DSDD for MSDOS $7.95 /box. Prime 41256 150 nanosecond memory chips only $2.25 each. Add $1.50 S/H. Dynamic Electronics, Box 896, Hartselle, AL 35640. (205) 773-2758

DISPLAY ADS

(Rate sheet 2 - March 1986)
Closing 1st of preceeding month.

Pages	1 time	2 times	3 times
*2	25	23	22
1	30	27	25
1/2	23	20	18
1/3	19	17	15
1/4	15	13	12

* We can use colored paper at no extra charge if ads are on both sides.

We can do ads in Red, Blue, or Brown. No all one color ads will be accepted. For color ads send artwork for each color. Add 40% for each color. Example: One page black and red for 3 times costs $25 + 10.00 = $35.00 each month.

Artwork must be camera ready and can be enlarged or reduced at no extra cost. We can do your ad at a nominal cost. Rates are per page or fraction per month. Enclose payment with ad copy. No X-Rated ads.

ADVERTISERS INDEX

We would appreciate it if you would let these advertisers know that you saw their advertisement in **Dynamic Color News**.

ATTENTION DEALERS

Are you taking advantage of our free services? We will list your new products and review them free. Also our ad rates are the lowest.

BULKRATE
U.S. POSTAGE
PAID
HARTSELLE, AL
35640
PERMIT NO. 21