

DYNAMIC COLOR NEWS

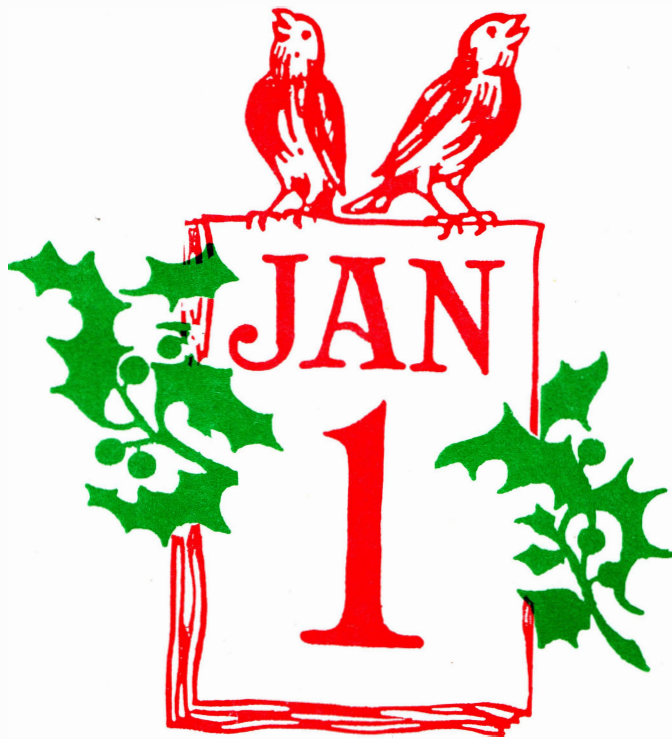
Radio Shack Color Computer
Magazine

Jan 1989
Issue #57

\$2.25

ASSEMBLY LANGUAGE

OS-9 & BASIC 09



C LANGUAGE

PROGRAMS

COCO 3

HAM RADIO

BASIC PROGRAMMING

RAMDISK

for the 512K COCO 3

A ramdisk operates similar to a disk drive except it is many times faster. The 512K ramdisk allows drives 2 and 3 to be ramdisks. You can backup a disk to either ramdisk or select either one for quick program or data loading. OS-9 is not required. A memory test program is also included. \$15

DYPRINT

Now you can print **LARGE** signs for special occasions 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.

MAXPRINT allows graphics to be blown up and printed on a standard printer. Any PMODE 4 picture can be printed. 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. Blow up pictures of friends and family generated by the DS-69B digitizer or make posters announcing sales or special events.

The **DYPRINT** package contains both **BANNER** and **MAXPRINT**. The cost is only \$19.95

NEW TERMINAL PROGRAM

DYTERM 2 - Allows a Color Computer to interface with **Modems, Terminals, or other Computers** using the ASCII port. 300-2400 baud, 1 or 2 Stop bits, 7 or 8 bit words, variable parity. Download programs from bulletin boards or other computers or upload your ASCII programs. Supports CoCo 2 and CoCo 3 Disk or Tape computers. Basic program with machine language sub-routines is easily modified.

Tape or Disk \$19.95.

DECIMAL ML ASSEMBLER

DISASM is a 6809 Assembler-Disassembler that allows machine codes to be assembled using English mnemonics & decimal arithmetic. It supports all 6809 codes and is especially useful for beginners. Learn Assembly programming without using hex. Disassemble machine language programs and print them to a printer. \$9.95

DS-69B DIGITIZER

Capture pictures from your VCR or video camera. Then print them on your graphics printer. Have your friends over for an evening of fun and digitize and print their pictures. Supports all color computers. The picture can be displayed on the COCO 3's high resolution screen. Pictures can be Labeled with **COCO MAX** and printed on a graphics printer or saved on disk. 256 x 256 resolution, 64 levels of grey, & 8 images per second. Plug in ROM pack requires a multipack expander. Works with all color computer disk systems.
DS-69B \$149.95 including shipping.

CC-THERM 2

CC-THERM 2 is a dual digital thermometer for Radio Shack Color Computers. It consists of two thermistors wired to the end of 10' and 20' flat cables for measuring inside and outside temperatures. The other end of the cable is wired to a joystick plug. The thermistors can be mounted on a wall, inside equipment, or outside for temperature measurements. Basic software on tape or disk continuously prints the temperature in both Fahrenheit and Centigrade. T or D software. \$19.95

CC=LT

Now you can measure both temperature and light. The joystick assembly includes a light and temperature sensor at the end of a 20' flat cable. Uses one joystick plug. T or D Software 19.95.

MEMORY MANAGER (for the Color Computer 2)

Did you know that the 64K Color Computer 2 and earlier computers have an extra 32K that is generally not used? Our **Memory Manager** allows basic or machine language programs to be run in either 32K bank. Banks are exchanged with an EXEC command. Also the second bank can be used as a ramdisk to store programs. This makes cassette operation faster than a disk. A third option configures the computer for the all ram mode allowing data or programs to be stored in the upper memory. The **Memory Manager** software is available on either cassette or disk. \$19.95.

MEMORY SAVER 2

Have you ever had a power failure or brownout to wipe out your program? The **Memory Saver II** is a battery backup assembly that prevents loss of programs due to power failures. It mounts under the keyboard and works with all color computers. Consists of gel rechargeable battery, control circuit, & miniature toggle switch. Will power a color computer for up to a couple of hours during a power failure.

Special sale price. \$29.95.

Add \$3 S/H. Specify Tape or Disk Software. Checks, VISA, & MC.

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

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.

Entire Contents (c) by DYNAMIC ELECTRONICS INC., 1989. DYNAMIC COLOR NEWS is intended for the private use of our subscribers and purchasers. All rights reserved. Contents of this magazine may not be copied in whole or in part without written permission from DYNAMIC ELECTRONICS INC. Subscriptions are \$18/yr for U.S.A. \$20 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, Inc. 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 TXT file.

Editor and Publisher
Bill Chapple W4GQC

Secretary - Dean Chapple
Artist - Tamara Moore
Contributor - John Galus
Contributor - Norm Matice
Contributor - Doug Canfield

CONTENTS

* Assembly Language (Part 2)	4
* COCO III (Part 6)	9
(High Resolution Text)	
* Air Attack (Game)	12
* Basic Programming	16
(Developing a ML Loader)	
* Machine Language Loader	19
* C Programming (New Series)	24
* New Products	26
* Math (Educational Game)	27
* Product Reviews	28
* Ham Radio & Computers	31
* VHF Antenna Design Program	32
* Editor's Comments	33
* Questions & Answers	34
* Auto-Exec ML Programs	36
* AUTOEXEC Program	37
# OS-9 & Basic 09	39
Cummulative Index	41
Bulletin Boards, Clubs, Advertiser's Index	42
* Included on Disk or Tape.	
# OS-9 Procedures are included on DCN on DISK.	

ASSEMBLY LANGUAGE

by
Bill Chapple

Part 2

Last month I introduced assembly language and explained why it is important to learn. An assembler is a program that writes machine language codes. These codes or instructions do not need translation and allow the computer to operate at its fastest rate. An assembler allows us to write instructions in a way that is easy for us. We write source code and it is translated or compiled into machine codes. Then the machine language program can be executed from basic.

All of us have loaded a machine language game or program using the (C)LOADM command from the keyboard. This command places a compiled program into memory so it can be executed. After the program is loaded we run it by typing EXEC and pressing the enter key.

COMMENTS

Comments are needed to allow us to keep up with what each section does. In basic we include remarks to tell what is happening. In assembly language we can include comments in our source file. An assembled program is difficult to troubleshoot because comments are not included to show what each section does. For example in basic we can write A=15. In assembly language we say load A with the value 15. In basic the value of A is placed in memory somewhere and the software keeps up with where it is located. In assembly language the programmer has to keep track of where the information is located.

There are two working registers which are the A and B registers. Each can contain a variable, but if we want more than two variables then they

will have to be placed in memory somewhere and we will have to keep up with the variables and their memory locations.

POINTERS

A pointer or vector tells us where to put information. We have the X and Y registers which are normally used for this purpose. Suppose we want to move a block of data from one memory location to another. The X register can point to the memory to get the data and the Y register can point to the location to store the data. One way to accomplish this is to load the register A with the value in memory indicated by X and then store the value in register A in memory indicated by Y. There are many operations that we can use on the X and Y registers. Increment allows us to increase the pointers value after an operation and decrement allows us to decrease its value after an operation.

GO TO & GO SUB

In assembly language we may wish to go to a subroutine to do a task. We use the terms BRANCH or JUMP to go to another memory location. We also have subroutines and logic tests similar to those in basic. We branch or jump to a subroutine. The branch instruction requires less memory for the instructions and is used for branches less than 127 bytes from our location. The jump command can take us anywhere in the memory map.

ARITHMETIC OPERATIONS

There are 3 arithmetic operations that we can use in assembly language. They are add, sub-

tract, and multiply. Some micro-processors do not have the multiply option, but we are fortunate in that it is included with the 6809 set of instructions.

ADDRESSING MODES

This was one of the hardest things for me to grasp when I started writing machine language programs so I want to expound on it a little. Let's consider the "A" register and look at ways that we can place a byte into the register.

IMMEDIATE MODE

In this mode the number we want to place into the register is contained in the next one or two bytes following the opcode. The term "opcode" means the operation code or instruction. If we want to load A with the value 137 then we would write the following:

LDA #137

This is similar to the basic statement $A=137$. The LDA means to load register A. The "#" means to use the immediate mode and the 137 is the decimal value to place into register A. This instruction only takes two bytes of memory. The first byte is for the opcode and the second byte is for the value. The A and B registers are 8 bit registers which can hold one byte.

To load the X register with 50000 we would enter the following:

LDX #50000

This would take 3 bytes with the first containing the opcode for LDX and the next two containing the value of 50000 in decimal.

Let's look at how a number this big can be placed into two bytes. A pointer or vector can point to any memory location

from 0 to 65535. The X and Y registers are generally used as pointers. To do this it is broked into a most significant (MS) byte and a least significant (LS) byte. The following basic program will do this conversion if we assume X is the value of the number.

MS=INT(X/256):LS=X-256*MS

The lower memory contains the MS and the upper memory contains the LS.

EXTENDED MODE

If we want to load register B with the value in memory location 31255 then we would enter the following assembly language command:

LDB 31255 or LDB >31255

The > is the operator sign but can be omitted with most assemblers.

EXTENDED INDIRECT

This is used when the address of a byte is stored in memory. Suppose memory 350 contains the address of a byte we want to place into the A register. Then we would enter

LDA [350]

The shift down arrow gives the left bracket and the shift right arrow gives the right bracket.

INDEXED MODE

This is the instruction that allows us to load the A register with the value in memory that the X register contains. To do this we would enter the following instruction:

LDA ,X

If we want to load A with the

value in memory X+5 we would enter

LDA 5,X

The offset is first and then the register.

If we want to increment or decrement a register after the operation then we can do the following:

- LDA 5,X+ Increase X by 1
- LDA 5,X++ Increase X by 2
- LDA 5,-X Decrease X by 1
- LDA 5,--X Decrease X by 2

For each of the preceding operations, the value of the X register is changed after the operation.

DIRECT ADDRESSING

Two bytes are required to cover the entire memory range. In direct addressing a register

called the direct page register is used to contain the MS byte. This means that we can use direct addressing by just changing the LS byte over a range of 256 memory locations. The direct page register is 0 unless set to a different value. To use direct addressing, the direct page register will be loaded with the MS byte. Then the LS byte is changed with the command

SSET DP=K

where K can be any number from 0 to 255.

HEXADECIMAL

Most assemblers use hex or hexadecimal numbers. These are to base of 16 instead of a base of 10 for decimal numbers. To indicate a hex number precede the number with a "\$" sign. \$FF is the same value as 255 but only requires two characters.

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 149⁹⁹.

REG. \$450

NOW \$149⁹⁹

★★THIS MONTH ONLY★★

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



- * 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

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



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!



	TAPE OR DISK	THIS MONTH ONLY
1 YEAR (12 Issues)	28.00	60.00
6 MO. (6 Issues)	16.00	35.00
1 ISSUE	2.00	8.00

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

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

The following basic commands will do conversions so you can get a feel for hex numbers.

30018 4
30019 3
30020 RTS 57

A=&H HEX NUMBER

B\$=HEX\$(DECIMAL NUMBER)

C=&HFF5

C\$=HEX\$(45352)

The letters A through F correspond to decimal numbers from 10 through 15. Hex and decimal numbers from 0 to 9 are the same.

EXAMPLE PROGRAM

I have always liked to include examples with my editorials. This month I want to use the LDA and STA commands. With these I will load a value into the A register and store it in memory. We will write a simple basic program to load the values into memory. This program will reside in memory starting at 30000 and will write the word "NEW" on the top left hand corner of the screen. The program is 21 bytes long and can be executed from basic by typing EXEC 30000. The chart below shows the assembly commands and values poked into memory:

MEMORY	COMMAND	VALUE
30000	LDA# 78	134
30001		78
30002	STA 1024	183
30003		4
30004		0
30005	LDA# 69	134
30006		78
30007	STA 1025	183
30008		4
30009		1
30010	LDA# 87	134
30011		87
30012	STA 1026	183
30013		4
30014		2
30015	LDA# 96	134
30016		96
30017	STA 1024	183

BASIC ML LOADER

The following basic program will load the values into memory starting at 30000. I used the READ- DATA method. The program can be run by typing EXEC 30000. Notice the word "NEW" at the top left of the screen when the machine language subroutine is executed.

```

10 ?~ML LOADER PROGRAM TO
15 ?~PRINT NEW ON THE SCREEN
20 FOR J=0 TO 30
30 READ A:POKE 30000+J,A
35 NEXT J
40 ?~ML PROGRAM IS LOADED~
50 ?~TYPE EXEC 30000~
60 DATA 134,78,183,4,0,134
70 DATA 69,183,4,1,134,87,183
80 DATA 4,2,134,96,183,4,3
90 DATA 57
    
```

HOLIDAY SPECIALS from SPORTSware

Offer good thru 31 Jan 89 Previous orders excluded

WARGAME DESIGNER: The LAST wargame you will ever need to buy. Play the 4 scenarios included. Modify them. Create your own 1 and 2 player simulations! Perfect for wargamers, adventurers, and science fiction addicts. See the RAINBOW AUGUST 88 review. You get 2 floppy disks & 23 page manual. Requires COCO 3 w/ 1 drive & RGB. CMP monitor or TV. Works great with ram disk. Regular \$29.00 Sale ONLY \$25.00.

STAND ALONE WGD SCENARIOS

INVASION NORTH: lead your modern forces across the river & capture 9 enemy held objectives.

ATTACK ON MOSCOW: Relive the WW2 German assault on Moscow.

ROBOT COMMAND: Enter the command post. Fight your way to the central computer and disable it.

DUNGEON WARRIOR: Rescue Jamie from her cell deep in Zarcos's dungeon.

GHOST HUNTERS: You and your team must clear the mansion of ghosts or die trying.

DESERT RATS: A free for all tank battle in the wastelands of North Africa.

ZULU REVENGE: A few hundred British soldiers face repeated assault by thousands of Africa's most feared warriors? You command!

All of the above are for 1 & 2 players. WGD is not necessary to play. WGD owners can modify these games with the WGD system.

Each of the above scenarios is only \$15.00 and comes w/ manual. Requires COCO 3 & 1 disk drive.

WGD ICON SETS: One disk full of Unit and Terrain icon sets for use with the WGD System. Just \$10.00

GRIDIRON STRATEGY: The BEST football strategy game ever written for the COCO 3. See the AUGUST 87 RAINBOW review. RGB CMP or TV. For two players. Unique playing system keeps the tension high! COCO 3 w/ 1 disk drive. Was \$21.00 NOW \$18.00 Set. Hut. Order!

BY POPULAR DEMAND: WEEKLY WINNER 2.0 our innovative lotto player's best friend is now available on disk and tape for both COCO 3 and COCO 2. Enhance your chances of winning your State lottery. For 3, 4, and 6 digit lottos with 1 to 50 number variations. All versions, tape or disk. ONLY \$15.00

All orders shipped **FIRST CLASS FREE** within 24 hours of receipt. **ORDER YOUR FUN TODAY!**

SPORTSware 1251 S Reynolds Rd Suite 414, Toledo, OH 43615 (419) 389-1515

FREE Catalog on disk (COCO 3 only) Send \$3.00 to cover cost of disk & mail. **SEE** before you **BUY**. Deduct \$3.00 from any product ordered from the floppy disk! Both sides full of value



COCO III

Part 6

by
John Galus

While the new hi-resolution text modes allows us to display characters on a graphic screen, it doesn't let us mix characters and graphics on the same screen. Here is where the Basic HPRINT command comes into play. With this command we can draw graphics pictures and label them on a hi-res screen. The format for the HPRINT command is as follows:

```
HPRINT(5,10),"DYNAMIC COLOR NEWS"
```

This will place the words DYNAMIC COLOR NEWS on the tenth line starting at the fifth position. You can even change the color of the letters using the HCOLOR command, depending upon which HSCREEN mode you are in. The HSCREEN2 or 1 mode gives you a 40 X 24 screen while, the HSCREEN 3 or 4 will give you a 80 X 24 text screen. One drawback of this HPRINT command is that when you print a character in one position on the screen, and then print another at the same position, the characters "merge" together leaving you with an unrecognizable character. After some disassembly of the HPRINT routine I discovered that doing the following poke will fix this.

```
POKE &HFOOC,&H12
```

This pokes a NOP code into a spot in the Basic ROM that for some unknown reason mixes the old and new character information together. One good feature about the HPRINT command is that the character set is in

memory starting at \$F09D and can be altered if you wish. For example, try this short Basic program.

```
10 FOR X=&HF09D TO &HF09D+7
15 POKE X,255:NEXT
20 HSCREEN2:HPRINT" A S P A C E ";
30 GOTO 30
```

Notice that the spaces will now be printed as solid squares. The HPRINT command can print characters from character 32 the space bar to character 128. Each character is 8 bytes long and is displayed differently depending upon the HSCREEN mode you are using. Try and write a program that can change and edit the characters, you should be able to come up with all sorts of interesting things.

Now let's look at a very new feature on the COCO III, the capacity to scroll the screen. This computer can change the horizontal or vertical size of a graphics screen. This means we can define a screen that is larger than the normal screen and then scroll the screen to display the unseen rest of the display. There are two registers that control this scrolling.

```
VERTICAL SCROLL REGISTER:
$FF9C
```

```
HORIZONTAL SCROLL REGISTER:
$FF9F
```

The vertical scroll register also uses the vertical offset registers located at \$FFF9D and \$FF9E as we shall soon see.

Normally, a graphic or text screen has a horizontal width of 160 or shorter. If we enable the horizontal scroll feature the screen is expanded to 256 bytes across. To turn on the horizontal scroll, bit 7 of the horizontal scroll register must be set, when it is cleared the horizontal size of the screen corresponds to the graphic or text mode that you are in.

The following is a table of the normal widths of the text and graphic modes available in Basic:

TEXT MODES

WIDTH	HORIZONTAL SIZE
32	64
40	80
80	160

HSCREEN

1	80
2	160
3	80
4	160

Once we have set the horizontal scrolling, we can move the screen right or left by placing values in bits 0-6 of the horizontal scroll register. A zero results in no movement. I wrote a Basic program that shows how to obtain this horizontal scrolling. This routine scrolls the words "DYNAMIC COLOR NEWS" across the screen. Notice how the letters move. The number of pixels that are scrolled depends on the mode you are in, 16 for a 2 color mode, 8 for a four and 4 for a 16 color mode such as we are using in the example below.

```

10 HSCREEN2
20 HPRINT(8,10), "DYNAMIC COLOR
NEWS";
30 S=128:POKE&HFF9F,S 'TURN
ON HOR SCROLL
40 I$=INKEY$:IF I$=~ THEN40
50 IF I$=CHR$(8) THEN IF S<180
THEN S=S+1
60 IF I$=CHR$(9) THEN IF S>128
THEN S=S-1
70 POKE &HFF9F,S:GOTO40

```

PROGRAMS

These are collections of programs from DYNAMIC COLOR NEWS. Number after program is the issue number.

DCN-1 * 64K all RAM, * 2- bank address file, Alarm Clock, Loan Interest, Character Generator, * Bank Switching.
* CC-2 Memory managers

DCN-2 Check Book Program., Ball Team Sort Program., Card Shuffling, Student Study Program, Address File.

DCN-3 Restore-Recover program lost after NEW command, Fast Food, Bar Graph, Memory Peek & Poke, Graphics draw.

DCN-4 Address File with Sort up to 100 names, Morse Code Generator, Star Constellations, Dueling Cannons.

DCN-5 COLOR COMPUTER 3 PROGRAMS CC-3 Memory Manager- Switch 8K blocks #38, CC-3 Error Trapping- Program to print error message #37, CC-3 Graphics #38, CC-3 Graphics Save #40

DCN-6 Accounts Payable- Business program #38, Dog Race (game) #40, Compound Interest- Figure best investment deal. #40, Address File Disk Sort (up to 100 names) #40, Invoice Program- Example for writing your own #36.

DCN-7 Meteors (game) #41, Graphics print- Use regular print for large picture #42, Parachute (game) #42, Music (Peace)- Hear quality computer music. #43, Geneology- Keep records of your family tree #39.

DCN-8 Oware (Game) #36, Save the Maiden (Word game) #43, Printer Utilities - Print information on screen to printer #44, Graphics Screen Dump Program #44.

DCN-9 Kingpede (game) #44, U.S. States #52, PMODE 4 to CC3 High Resolution converter #45, Logic (ed game) #47, Fast Dir (disk pgm) #48, Guns (game) #49.

DCN-10 Frequency Counter #45, Superspell (ed game) #49, Basketball Math #49, FANTASY2 Musig pgm #45, Kwik Kopyur (128K CC3 disk pgm) #51, Loan pgm #52.

DCN-11 TRIO (3 pgms) #52, Mysterious Island #50, Player Guitar #51, Teacher Grade pgm #52, EDT-MAS (AL UT) #52, Audio Generator #44, Bustout #51.

Programs are \$5.95 each tape or disk. Add \$1 shipping. Checks, VISA & MC.

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

Use the right and left arrow keys to scroll the letters back and forth across the screen. Vertical scrolling is done a little differently. As we have seen the horizontal scroll can move at best 4 pixels per scroll. We can scroll more smoothly vertically at a rate of one line per scroll increment. We have seen a form of coarse vertical scrolling when letters on a low resolution text screen move up the screen. Now we can make the information on the screen move up or down one scan line at a time. This vertical scrolling is enabled by clearing bit 3 of the vertical scrolling register at \$FF9C. Each line in a text mode can be scrolled by storing a value from 0-7 in bits 0-2 in this same register. This allows moving up or down one video screen scan line at a time. When the scroll value has reached 7, the process is completed.

To continue to scroll the screen up we must set the scroll register back to zero and then increment the value held in the Screen Offset Registers at \$FF9D and \$FF9E. We do the inverse of this procedure to scroll vertically down. Here is a Basic example of this fine text screen vertical scrolling:

```

10 WIDTH40
20 LOCATE0,0:PRINT~VERTICAL
  SCROLL~;
30 POKE &HFF9C,0 'TURN ON
  VER SCROLL
40 I$=INKEY$:IF I$=~" THEN40
50 IF I$=~"U" THEN V=V+1:IF V>7
  THEN V=0
60 IF I$=~"D" THEN V=V-1:IF V<0
  THEN V=7
70 GOTO 30
    
```

Vertical scrolling on a hires screen is done by adding the screen's length to the vertical screen position for each line scrolled up. Here's an example.

```

1 HSCREEN2
2P=PEEK(&HFF9D)*256+PEEK(&HFF9E)
20 POKE&HFF9C,0 'TURN ON VER
  SCROLL
30 HPRINT(10,10),"HI SCROLL~";
40 IF INKEY$=~" THEN40
50 POKE&HFF9D,P/256
55 I=INT(P/256)*256
60 I=P-I:POKE&HFF9E,I
70 P=P+40 'UP ONE LINE
80 GOTO40
    
```

Experiment with these routines, try and put them together or have the joystick control there movement. In the next part of this series we will take a look at the COCO III's colors and other interesting features of this powerful new computer.

Your Personal Electronic Photo Album

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

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

- 8 MGE format files (for ColorMax3) or
- 10 CM3 format files (for CoCoMax3) or
- 15 PHODE format files (for CoCoMaxII, MCPAINT, etc.)

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

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

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

RainyDay Software

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

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



AIR ATTACK

This is an exciting one player machine language game that requires a joy stick. The object of the game is to drop bombs on ground targets and shoot down missiles, flying saucers, and fireballs. The joy-stick controls your movement. This program is provided as a courtesy of T&D Subscription Software (See their advertisement on page 6) and is used by permission.

The following basic program will load the program. To use it, upper memory must be reserved to load the program. Do the following:

```
POKE 25,80:POKE 80*256,0: NEW
```

Then load the program and make the machine language save to a cassette or disk.

```

2 CLS
6 PRINT
8 PRINT:PRINT~STANDBY WHILE MACH
  INE LANGUAGE PROGRAM IS BEIN
  G GENERATED~:PRINT
10 BE=7935:M=BE
12 READ X$
14 IF X$=~@~ THEN 40
16 L=LEN(X$)
18 FOR J=1 TO L STEP 2
20 A$=MID$(X$,J,2):B=PEEK(M)
22 C$=LEFT$(A$,1):D$=RIGHT$(A$,1
  )
24 X=ASC(C$):Y=ASC(D$):X=X-48:Y=
  Y-48
26 IF X>9 THEN X=X-7
28 IF Y>9 THEN Y=Y-7
30 V=16*X+Y:POKE M,V
32 M=M+1:IF M=EN THEN 40
34 NEXT J
36 PRINTM
38 GOT012
40 PRINT~DATA IS TRANSFERRED
42 PRINT~1 SAVE ML PGM TO DISK
44 PRINT~2 SAVE ML PGM TO CASSET
  TE
46 INPUT X
48 'PUT PGM NAME HERE X$
49 EN=M:EX=BE
50 IF X=1 THEN SAVEM X$,BE,EN,EX
52 IF X=2 THEN CSAVEM X$,BE,EN,E
  X
60 END
100 DATA 86FFB7FFC7B7FFC8B7FFCD1
  7005D170D3886C8B7FF22B7FFC517
  00851700E3AD9FA00027FA20D9347
  6108E0008ECC1ED81A6C0A7803088
  1D313F26F13576393476108E0004E
  CC1ED81ECC1ED8130881C313F26F1
  3576393476108E0008ECC1ED81308
  81E
110 DATA 313F26F53576398600B7FF2
  2B7FFC48E120086A0C6A0ED808C14
  0025F9CE2BCEAEC1ECC11083FFFF2
  704ED8120F4EC401083FFFF26EAD
  9FA00027FA394F5F8E1200ED818C1
  E0225F9CE28B88E1684C604F72782
  17FF6C34403005CE2B2A17FF62354
  030
120 DATA 89017B33C8607A278226E58
  E1690CE2A3817FF4B8E1810CE2A98
  17FF428E1695CE2B4217FF398E181
  517FF338EFFFF301F26FCAD9FA000
  27FA394F5FFD272EFD2730B727328
  604B7276B8619B7278F8E1200CC00
  00ED818C1CC025F9CC5555ED818C1
  E02
130 DATA 25F9CCFFFF8E12A0ED818C1
  2C025F98E1823BF276D860CB7277D
  8631B7277E4FB7276FCE287817FEE
  A8E120CB6276B4A270817FEDE301C
  4A26F88601B72770B72777B727878
  627B727738E2824BF27717F27758E
  27904F5FED818C280A25F9C607F72
  782
140 DATA 8E121E108E272CA6A0C6053
  DCE2AF833CBC605A6COA700308820
  5A26F63089FF5F7A278226E1FE277
  17A27752C2C8603B727757A277026
  18A6C0B7277AE6COF727701183287
  42503CE2824FF2771200AB62773BB
  277A4AB72773F62773862003D8E17E
  030
150 DATA 8B108E281CB62775BB27753
  1A6C620F02770F727828627BB2773
  B72774F62770B6277A271B8101272
  91F98BB2774B72774A621A7003088
  215A26F83088E02013A620A700308

```

8E17A27745A26F530882020013AFE
 2771A6COE6COF127822506F62782F
 727
 160 DATA 77F727768100271B8101272
 91F98BB2774B72774A621A7003088
 215A26F83088E02013A620A7007A2
 7743088E15A26F530882020013AF6
 2782F02776F7278226B1BF2778FF2
 785AD9FA00ABE276DB6015A811025
 2C81302549B6277D4C814B271EB72
 77D
 170 DATA 7C276FB6276F810425114FB
 7276FA700A78820A78840A7886030
 012021B6277D4A271BB7277D7A276
 F2C138603B7276F5FE703E78823E7
 8843E78863301FB6015B811025128
 130251C4F5FED00ED023088207C27
 7E200E7A277E4F5FED8860ED88623
 088
 180 DATA EOCE2878B6276FC6103D33C
 B17FD10BF276DB6276FC6043DCE28
 OC33CBEC88E0A4402629E4412625E
 C88E2A442261EE443261AEC890080
 A4402612E441260EEC890082A4422
 606E443260220058601B72789B6FF
 0184F7B7FF01B6FF0384F7B7FF03B
 6FF
 190 DATA 238A08B7FF23BE2778FE278
 5A65E8101102600D4B62777810410
 2500CB7A2783102600C48610B7278
 3B627842740810127707F2784C603
 108E27906D20270831265A26F7160
 0A17A27872C058603B72787B62787
 C6603DCE28B833CBEF218674A7203
 089
 200 DATA FEFDAF238603A725207A860
 1B72784C603108E27A2A620270731
 285A26F720648601A7208674A721B
 62774A7223089FEFADF238603A725
 CC2A38ED262046B6278A810325057
 F278420058602B727847D278A2730
 108E27BA6D20270E31286D2027083
 128
 210 DATA 6D202702201A8601A720867
 4A721860AA7228E135DAF238603A7
 258E2A98AF26C603F72782108E279
 06D20272A6A20260ACE27F2AE2317
 FBA2201C6A252C0A8603A725AE233
 01FAF23EE21A625C6183D33CBAE23
 17FB8431267A278226CB108E27A2C
 603
 220 DATA F72782A620102700636A212
 60C6F20AE23CE27F217FB6020536A
 252C0AAE23301FAF238603A725A62
 081022716A622B0277EB72781A621
 B0277DB127812E1F8602A7206A22A
 6228106220C6F20AE23CE27F217FB
 222015AE233088E0AF23AE23EE26A
 625
 230 DATA C6183D33CB17FB0B31287A2
 7821026FF8EC603F72782108E27BA
 6D20274C6A21260C6F20CE27F2AE2
 317FAE7203CA6208101270FAE2330
 88E06A22A622810A22116A20AE233
 088206C22A622812F25026C206A25
 2C068603A725301FAF23C618A6253
 DEE
 240 DATA 2633CB17FAA931287A27822
 6A97A278826658605B72788B6FF00
 81FF2759817F2755C604108E27D26
 D20270731245A26F72044B6277D8B
 10A720BE276D308843AF22B6277E4
 C4CA7217C278C7C278BC604108E27
 E26D20270731245A26F72017BE276
 D30
 250 DATA 8900A3AF22B6277D8B0CA72
 0B6277E8B05A721C604F72782108E
 27E2CE27D26D402730AE424FA700A
 6012711A640B72780A641B7277F6F
 4017011C20163001AF42A6408B048
 17C25046F402006A7408603A7006D
 202741AE226F006D8840272CA6882
 184
 260 DATA 0F3404E6882158585858E78
 820AA88203504A78820A620B72780
 A6214C4CB7277F1700CF6F20200C3
 08820AF226C2186FOA78820312433
 447A27821026FF7CB6278B274C810
 1272BF6278D54F72782C614CE2BBA
 A6C0B7FF205A26F8C61C1212125A2
 6FA
 270 DATA 7A278226E67A278C26317F2
 78B202CF6278DF72782C614CE2BBA
 A6C0B7FF205A26F87A278226EE7A2
 78B200F8619F6278D3D1F01121212
 301F26F9B6278A81022515B6278DB
 1278E270D7A278F26087A278D86FF
 B7278F12B627898101270316FAB01
 700
 280 DATA 0B7F27897A276B1026FA043
 9CE27F233C907D0108E07D0BE276D
 17F91AA64084FCB7FF20335F313F2
 6F0393456CE2790C603B62780A140
 2523800CA140221D3404C6FF8602B
 7278B8608B7278C35048E2B5AAF41
 8608A7408D50204B33465A26D1CE2
 7A2
 290 DATA C606B62780A1412535800CA
 141222FB6277FA14225288008A142
 22228E2B5AAF46B6278A81012E048
 608A7411700171700148602B7278B
 8608B7278C200533485A26BF35563
 93476C605108E272E8E121C6C20A6
 A081092204C60120036F3F4FCE2AF
 834
 300 DATA 04C6053D33CBC605A6C0A70
 03088205A26F635043089FF5F5A26
 D2B6278A270A810127108102271A2



ARES SOFTWARE PRESENTS

DRBUS 1 : Elements of fury. fantasy adventure role playing system set in the land of Asirica filled with monsters , magic, warriors and magicians. You create a party of four characters to search for the legendary RINGS OF ELEMENTAL CONTROL. 12 character roster, Individual player combat, One or two player game, Hires animated graphics.

64k CC2 or CC3 joystick/disk..... \$25

BLUE AND GRAY : A grand strategy simulation of the entire Civil War. You can control the armies of North or the South to recreate the battles of America's bloodiest War! 16 color graphics, full game save, four levels of play, stragtic map. Hires scrolling game board.

CC III only 128k disk 100% ML..... \$25

RED ALERT: A starship combat simulator. Object of the game is to defeat the computer controlled enemy vessel by using your ship's capacities strategic maneuvers, and your own smarts. Over 27 ships to choose from or design your own. CC 64k joystick/disk..... \$23

WAR AT SEA: Wooden ships simulate ship to ship battles during the 18th century. Player controls a number of sailing ships from different nations and must pit his seamanship against the computer or another player. CC 64k joystick/disk 100% ML..... \$23

TASK FORCE: Modern Naval Carrier Warfare in the Mediterranean and Middle East. You can control squadrons of modern naval warplanes.

CC 64k joystick/disk..... \$23

DESERT FOX: Rommel's North African campaign. CC 64k 100% ML/disk \$22

FIRE AND STEEL: Napoleon's return, Waterloo Campaign. CC 64k disk 100% ML..... \$22

FINAL FRONTIER: War in Space. Build fleets of starships to defeat the Volsung empire 32k disk.. \$20

STALINGRAD: The turning point in the war in Russia 64k disk 100% ML..... \$22

REDSTAR: Operation Redstar NATO vs. WARSAW PACT 32k disk..... \$20

DARKHORSE: Operation Darkhorse sequel to Redstar. Invade the U.S.S.R 64k disk 100% ML..... \$22

PHALANX: Alexander the Great, the world's greatest Conqueror 32k disk 100% ML..... \$15

SAGA: Save the beautiful princess in this Graphic adventvre 32k disk 100% ML..... \$15

INFERNO: Save the world from total destruction Graphic adventure 32k disk 100% ML..... \$15

All prices include shipping and handling. Foreign orders add \$3.00. New York residents add sales tax.

Send payments to :

JOHN GALUS
55 Wilkesbarre Ave.
Lackawanna, New York 14218
(716) 823-3144

```

024B62730271F7C278A201AB62730
810325138602B7278A200CB627308
10525058603B7278A357639202020
202053434F52453A2000005858585
834
310 DATA 484947482053434F52453A2
00000000000000053414D45204741
4D453F202859204F52204E2920204
C4556454C3F2028312D392920C002
308C00A68595962705C61517010E9
A96979617E94F2707812C01D017E9
8304000000012609C6151700F220F
20D
320 DATA 961026FD8817E97820EE061
020404080020401088D108C8D100C
9316FF258D068C8D0616FD0586108
C86119794391700A9D79517FE8524
0317FE2D8603979D0C9339170095D
D958602979D20A432628603979D17
E8E1272D978617E8DA27269186102
701
330 DATA CC9795860197930D822B0A1
701EE860797FFFFFF003FFFFFFC00F
FFFFFF003FFFFFFC550015400550015
4011F001001060210001501150118
02150005011502050115011F00070
11402070110011F011F0003011F00
03011F0003011F0003011F000E011
F02
340 DATA 03011F0203011F0203011F0
203010A020E00090209011F011F00
00A0002A02A8000AAAAA002AAA8C0
0000028000A80AA0002AAAA800AAA
A00000000A0002A02A8000AAAAA00
2AAA8000000028000A80AA0002AAA
A800AAAA003FFC00CA8300C80300C
A03
350 DATA 00C80300FFFF00EEEE00BBB
B000FFF0032A0C03200C03280C032
00C03FFFC03BBB802EEEC003FFC00
CA8300C80300CA0300C80300FFFF0
0EEEE00BBB000FFF0032A0C03200
C03280C03200C03FFFC03BBB802EE
ECOFF000300C00C00300C003003FF
C00
360 DATA 0AA000300C000C300003FC0
00EAB003AAAC03AAAC00FFF0003FC
000C03003000C000FF0003AAC00EA
AB00EAAB003FFC000FF000300C000
C300003FC000C03003000C03000C0
0FFF0003FC000C03003000C000F00
0C03000C0F00030C0000FC0000F30
0CF
370 DATA C300FC33000003C00000C00
303C000C300003F00003CC033F0C0
3F00C00000F000003000C0F00330C
0000FC0000F300CFC300FC3300000
3C00000C00303C000C300003F0000
3CC033F0C03F0CC000000C0000030

```

```

00000C30000FFC002AA8008002002
AA8
380 DATA 0000000003000003000003
00003FC000AAA0008002002AA8000
00000000300000C000C30003FF000
2AA8008002002AA800000000000C00
000C00000C00003FC000AAA002008
000AAA00300000380000300000FC0
00CFCC00FFFC00CFCC0000000000C
000
390 DATA 00C00000C00003F00033F30
03FFF0030F30000000000300000B0
0000300000F8000CFCC00FFFC00CF
CC0000000000C00000C00000C0000
3F00033F3003FFF0030F300000000
000000FF0003AAC00EAAB003AAC00
OFF000000000000000000000003FC0
00E
400 DATA AB003AAAC00EAAB0003FC000
00000000000000000000FF000300C0
0C00300300C000FF00000000000000
0000000003FC000C03003000C00C0
30003FC000000000000003F3333333
F3C0C0C0C3F3F030C303F3F033F03
3F30303F0C0C3F303C033C3F303F3
33F
410 DATA 3F030C30303F333F333F3F3
33F03033C3F3F0C33330C33330C33
333F3F3F00000000000000000003F3
F3F0333330C33333033333F3F3F00

```

MEMORY MANAGER
for
The COLOR COMPUTER 2

Did you know that the 64K Color Computer 2 and earlier computers have an extra 32K that is generally not used? Our Memory Manager allows basic or machine language programs to be run in either 32K bank. Banks are exchanged with an EXEC command. Also the second bank can be used as a ramdisk to store programs. This makes cassette operation super fast because programs can be saved to the ramdisk. A third option configures the computer for the all ram mode allowing data or programs to be stored in the upper memory. Our Memory Saver 2 allows your programs to be saved even with power failures. The Memory Manager Software is available on either cassette or disk.

\$19.95 +\$2 ship.

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

BASIC PROGRAMMING

In this series we are showing how to write basic programs. For the past few months we have been concentrating on disk files and commands. This month we want to look at loading machine language programs from basic.

STRINGS

This is a new command that we will be using. Our procedure will be to build a basic program that will contain the data for the machine language program. This will be "merged" with a basic loader program. The `STRING$` command converts a number to a string. We will use this for line numbers for the file that we will be building. Our data will start at line number 100 and increase in increments of 10 until all of the data has been converted. `STRING$` is used as follows:

```
L$=STRING$(100)
```

After basic executes this command then `L$="100"`.

DATA STATEMENTS

If we let `X` contain the line number, then we can create the first part of a data statement with the following:

```
16 X=100:M=BE
20 X$=STR$(X)
25 P$=X$+" DATA "
```

Notice that `P$` will be the line number plus the word "DATA".

Our procedure will be to construct a file consisting of DATA statements for the machine language program. All machine language programs require a begin-

ning (BE), ending (EN), and execution address. These will have to be included in our program. The variables BE and EN will be used to designate the beginning and ending of the machine language data.

HEXADECIMAL

Our program will carry the data as HEX characters which have a base of 16. The following command converts decimal to HEX characters.

```
A$=HEX$(A)
```

The program will use two HEX characters for one decimal number. The `A$` from the previous equation can have one or two characters. We want to make the programming easy so our program will add the extra character for values less than 16. We can check the length of a string by using the `LEN(A$)` command. If it is 1 then we will add a "0" to the left of the string and create a new string. The following line does this:

```
44 IF L=1 THEN A$="0"+A$
```

ASCII DISK FILE

As stated earlier, we will create a new disk file that will contain the data in ASCII form. Then each piece of data can be printed to the disk file as it is generated. For example the line number and word DATA can be printed to the disk as it is created. Also the strings for each data element can be printed to the disk as they are created. If the file is #1 then to print to it we use the following:


```
52 PRINT #1,A$;
```

Notice the semicolon is used to print the data next to the previous data.

FOR-NEXT LOOP

These powerful loops allow us to complete our data statements. If we want each statement to contain 100 bytes then a FOR-NEXT loop can do this for us:

```
36 FOR J=0 TO 99
40 A=PEEK(M+J):A$=HEX$(A):
  L=LEN(A$)
44 IF L=1 THEN A$=~0~+A$
52 PRINT #1,A$;
56 NEXT J
```

Notice how simple the programming is with the FOR-NEXT loop. Lines 36-56 complete a data statement on disk. A carriage return will be required at the end of the line. The statement number needs to be increased by 10, and the memory increased by 100 before starting the next line. The following statements do this:

```
60 PRINT #1,CHR$(13)
64 X=X+10: M=M+100: IF M>=EN
  THEN 72
68 GOTO 20
```

The conditional statement in line 64 causes the program to go to 72 if all of the data has been processed. Line 72 will close the file and end the program. Line 68 returns to line 20 to start a new data statement.

DATA GENERATOR PROGRAM

The following program generates a disk basic program that contains the data for the machine language program.

```
2 PRINT~THIS IS A MACHINE
  LANGUAGE CONVERSION PROGRAM
  TO SAVE MACHINE LANGUAGE DATA
```

```
ON DISK.
```

```
3 INPUT~ENTER BEGINNING (16128
  FOR MUSIC~;BE
4 INPUT~ENDING~;EN
8 INPUT~ENTER NAME~;N$
12 OPEN ~O~,#1,N$
16 X=100:M=BE
20 X$=STR$(X)
24 P$=X$+~ DATA ~
28 A$=~:B$=~
32 PRINT#1,P$;
36 FOR J=0 TO 99
40 A=PEEK(J+M):A$=HEX$(A):
  L=LEN(A$)
44 IF L=1 THEN A$=~0~+A$
48 PRINTJ+M;A;A$
52 PRINT#1,A$;
56 NEXT J
60 PRINT#1,CHR$(13);
64 X=X+10:M=M+100:IF M>=EN
  THEN 72
68 GOTO20
72 CLOSE #1:PRINT~END OF
  DATA~:END
```

USED ORIGINAL SOFTWARE

Cat. No.	Name	Price
COCO 2		
Tandy		
26-3109	Color Scripsit II (PP)	9.95
26-3296	Interbank Incident (D)	9.95
26-3059	Backgammon (PP)	7.95
26-3291	VARLOC (Game) (D)	9.95
26-3064	CYRUS-World Chess (PP)	9.95
INFOCOM		
	Enchanter (Adventure Game) (D)	9.95
	Infidel (Adventure Game) (D)	9.95
	Witness (Adventure Story) (D)	9.95
Pickly-Pear Software		
	Hall of the King (2 disk adv. game) (D)	9.95
COCO 3		
Tandy		
26-3262	Deskmate 3 (D)	49.95
26-3271	Microscopic Mission. (D)	9.95

(PP)= Program Pak, (D)= Disk, Shipping included in price. Indicate second choice. Checks, Visa or MC cards.

Dynamic Electronics Inc.
P. O. Box 896 (205) 773-2758
Hartselle, AL 35640

THE LOADER PROGRAM

The previous program just allowed us to convert the machine language data into DATA statements. Next we need to write a loader program and MERGE the data program with it. The "MERGE" command allows two programs to be combined. The program that is to be merged must be saved in ASCII format. The data we created in the previous program saves the results in ASCII.

HEX-DECIMAL

Remember that we created data statements using 2 HEX characters for each byte. We need a procedure to convert this back to decimal so we can poke the value into memory. The following routine will do this for us:

```

22 C$=LEFT$(A$,1):D$=RIGHT$(A$,1)
24 X=ASC(C$):Y=ASC(D$):X=X-48:Y=Y-48
26 IF X>9 THEN X=X-7
28 IF Y>9 THEN Y=Y-7
30 V=16*X+Y

```

The ASCII for numbers starts at 48 for a 0. Therefore the 48 is subtracted in line 24. Letters start at 65 with an A. An A represents 10 in HEX so a 7 has to be subtracted in lines 26 and 28 for values greater than 9. Line 30 calculates the value to be poked into memory.

The program of course must not occupy the area that the machine language program uses. The PCLEAR commands can be used to move the program up or down in memory. Locations 25,26 and 27,28 shown the beginning and ending of basic. The following formulas can be used to determine these.

```

BE=256*PEEK(25)+PEEK(26)
EN=256*PEEK(27)+PEEK(28)

```

Put the beginning of the data in line 10. This is the memory in which the first byte is to be

poked. Next MERGE the data program. If the data program is called "DATA" then type MERGE DATA. After the merge, the program will be the desired machine language loader.

ML LOADER

```

2 PCLEAR 4:CLS
3 PRINT"ML LOADER COPYRIGHT (C) 1988
4 PRINT"BY DYNAMIC ELECTRONICS INC
5 'PCLEAR IN LINE 2 MAY NEED TO BE CHANGED TO LOCATE PROGRAM IN DIFFERENT MEMORY FROM THIS PGM.
6 PRINT
7 'DATA SHOULD BE MERGED WITH THIS PROGRAM AND START IN LINES 100. AN @ MEANS END OF DATA
8 PRINT:PRINT"STANDBY WHILE MACHINE LANGUAGE PROGRAM IS BEING GENERATED":PRINT
10 'LIST MEMORY FOR START OF DATA HERE: EXAMPLE BE =30000
11 M=BE
12 READ X$
14 IF X$="@" THEN 40
16 L=LEN(X$)
18 FOR J=1 TO L STEP 2
20 A$=MID$(X$,J,2):B=PEEK(M)
22 C$=LEFT$(A$,1):D$=RIGHT$(A$,1)
24 X=ASC(C$):Y=ASC(D$):X=X-48:Y=Y-48
26 IF X>9 THEN X=X-7
28 IF Y>9 THEN Y=Y-7
30 V=16*X+Y:POKE M,V
32 M=M+1:IF M=EN THEN 40
34 NEXT J
36 PRINTM
38 GOTO12
40 PRINT"DATA IS TRANSFERRED
42 PRINT"1 SAVE ML PGM TO DISK
44 PRINT"2 SAVE ML PGM TO CASSETTE
46 INPUT X
48 'PUT PGM NAME HERE X$
49 EN=M:EX=BE
50 IF X=1 THEN SAVEM X$,BE,EN,EX
52 IF X=2 THEN CSAVEM X$,BE,EN,EX
60 END
50000 DATA @

```

PUBLIC DOMAIN SOFTWARE

<p>* PD-1 GAMES</p> <p>MENU BAS 0 B 1 BEAST BAS 0 B 1 BEAST DAT 1 A 1 BOBO BAS 0 B 3 GUNNER BAS 0 B 2 HOW BAS 0 B 3 LANDER BAS 0 B 3 LIFE BAS 0 B 3 MAX BAS 0 B 3 POKER BAS 0 B 2 BIGRITHM BAS 0 B 3 BLACKBOX BAS 0 B 2 BLOCKADE BAS 0 B 1 BUSJUMP BAS 0 B 1 CHUTE BAS 0 B 2 GO BAS 0 B 3 HANGMAN BAS 0 B 2 OHELLO BAS 0 B 2 TARTUS BAS 0 B 1 TARTUS2 BAS 0 B 1</p> <hr/> <p>* PD-2 GAMES</p> <p>MENU BAS 0 B 1 RUBIC BAS 0 B 5 FRACTAL BAS 0 B 1 KALSCOPE BAS 0 B 2 TARTUS BAS 0 B 1 TARTUS2 BAS 0 B 1 WORLD3D BAS 0 B 4 LIFE BAS 0 B 2 ADVENT BAS 0 B 4 ADVENT DOC 1 A 2 HURKLE BAS 0 B 2 REVERSE BAS 0 B 2 GUESSFR BAS 0 B 2 SCRAMBLE BAS 0 B 3 PIZZA BAS 0 B 2 CINQUAIN BAS 0 B 2</p> <hr/> <p>* PD-3 GAMES</p> <p>MENU BAS 0 B 1 AANDAN BAS 0 B 2 STARTREK BAS 0 B 9 TREKINST BAS 0 B 3 SEQUENCE BAS 0 B 2 ALPHAEET BAS 0 B 3 GEOGRAPH BAS 0 B 4 FLASH BAS 0 B 4 BAGELS BAS 0 B 3 OREGON BAS 0 B 9 MULTIPLY BAS 0 B 2</p> <hr/> <p>* PD-4 ML GAMES</p> <p>MENU BAS 0 B 1 PONG BIN 2 B 1 SQUASH BIN 2 B 2 BLOCKADE BIN 2 B 2 GERM BIN 2 B 1 WIGHORM BIN 2 B 2 GRID BIN 2 B 2 ZEROG BIN 2 B 2 3DTICTAC BIN 2 B 7 HOPBOP BIN 2 B 5 ICEWAR BAS 0 B 6 CIVILWAR BAS 0 B 4 TICTACTO BIN 2 B 7</p> <hr/> <p>* PD-5 GAMES</p> <p>MENU BAS 0 B 1 CAVE BAS 0 B 4 WARGAME BAS 0 B 2 WARGAME BIN 2 B 1</p>	<p>WARGAME2 BAS 0 B 5 WARROOM BIN 2 B 3 NORAD EAS 0 B 3 ANDREA BAS 0 B 5 CURSE EAS 0 B 4 GARGOYLE BAS 0 B 6 KINGTUT BAS 0 B 7 TAIPAN BAS 0 B 6</p> <hr/> <p>DSK-6</p> <p>SPELL & FIX- FIND SPELLING ERRORS IN TXT DISK FILES</p> <p>MENU BAS 0 B 1 MANUAL TXT 1 A 12 SPELLFX2 BAS 0 B 1 SPELLFX2 BIN 2 B 6 SPELLFIX EAS 0 B 1 DICT TXT 1 A 33 COREDICT TXT 1 A 1 SAMPLE TXT 1 A 1 BUILD EAS 0 B 1 LIST BAS 0 B 1 RESET BAS 0 B 1 APPEND BAS 0 B 1 ADDWORDS BIN 2 B 3</p> <hr/> <p>PD-7 DISK UTILITIES</p> <p>MENU BAS 0 B 1 BASIC64 BIN 2 B 1 BSEARCH BIN 2 B 1 DISKCOMP BIN 2 B 1 DISKTEST BIN 2 B 3 DISKWASH BAS 0 B 1 DOS64K BAS 0 B 2 DSDBOOT BIN 2 B 1 LIST BIN 2 B 2 PRINT BIN 2 B 3 PRINTDIR BAS 0 B 1 RECOVER BIN 2 B 1 ROMBACK BAS 0 B 1 ROMFIX BIN 2 B 1</p> <hr/> <p>PD-8 DISK UTILITIES</p> <p>SCRN51 BAS 0 B 1 SCRNS1 BIN 2 B 1 SCRNDemo BAS 0 B 2 SDC BIN 2 B 1 SQUEEZE BIN 2 B 1 SSDBOOT BIN 2 B 1 TAPE2DSK BAS 0 B 1 TIMER BIN 2 B 2 UNLOCK BIN 2 B 1 BACKUP BIN 2 B 1 BACKUP1 BIN 2 B 1 MORE BIN 2 B 3 SPEAK BIN 2 B 3 PCLEARFX BIN 2 B 1 MULTBACK BIN 2 B 1 MULTBACK DOC 1 A 1</p> <hr/> <p>PD-9</p> <p>TERMINAL PROGRAMS</p> <p>MENU BAS 0 B 1 TELETERM BIN 2 B 3 TELETERM CAS 2 B 3 TTHELP DAT 1 A 4 MTERM BIN 2 B 6 MTERM VIP 1 A 19 MICONFIG BAS 0 B 3 MTERM+ BIN 2 B 6 DATATRDE BIN 2 B 3</p>	<p>KERMIT BAS 1 A 1 KERMIT BIN 2 B 2 HAYESAE BIN 2 B 4 HAYESAE DOC 1 A 6</p> <hr/> <p>PD-10</p> <p>COLOR COMP. FORTH</p> <p>MENU BAS 0 B 1 FORTHMAN UL1 2 B 7 FORTHMAN UL2 2 B 7 FORTHMAN UL3 2 B 1 FORTH BIN 2 B 3 EDIT DAT 1 A 3 FRTHDOC1 TXT 1 A 7 FRTHDOC2 TXT 1 A 7 FRTHDOC3 TXT 1 A 1 FRTHDOC4 TXT 1 A 7 32KFORTH BIN 2 B 4 NEWFORTH BIN 2 B 3 WE BAS 0 B 1</p> <hr/> <p>PD-11 MCPAINT</p> <p>A COMPLETE GRAPHICS DEVELOPMENT PROGRAM WITH INSTRUCTIONS</p> <p>RUN-ME BAS 0 B 1 MCPAINT BIN 2 B 11 ICONS SYS 2 B 3 MCDOC DOC 1 A 11 PRINTDOC BAS 1 A 1 GLASDEMO BIN 2 B 6 STARS BIN 2 B 2 1940S SET 2 B 1 BLOON SET 2 B 1 BOLD SET 2 B 1 FANCY SET 2 B 1 GREEK SET 2 B 1 GREEKU SET 2 B 1 HEBREW SET 2 B 1 OLDENG SET 2 B 1 TYPING SET 2 B 1 EPSON DRV 2 B 1 EPSON2 DRV 2 B 1 ANIMATE BAS 0 B 1 ANIMAT BIN 2 B 1 BANNER BAS 0 B 2 MCUTIL BIN 2 B 1</p> <hr/> <p>* PD-12</p> <p>PMODE 4 PICTURES</p> <p>CHURCH, ROSES, RUN"PIXFILES"JOY- STICK IS REQUIRED</p> <p>XIXCMP BAS 0 A 3 OUTPOST BAS 0 A 3 OUTPOST BIN 2 B 3 SFIELD BAS 0 A 2 SFIELD BIN 2 B 3 PIXFILES BAS 0 B 3 TRUCK BIN 2 B 3 MODEM BIN 2 B 3 HORSE BIN 2 B 3 MISSION BIN 2 B 3 CLOISTER BIN 2 B 3 RAIN BIN 2 B 3 EAGLE BIN 2 B 3 ROSES BIN 2 B 3 CHURCH BIN 2 B 3 GARDEN BIN 2 B 3 PRES BIN 2 B 3 LONI4 BAS 0 A 3</p>	<p>PD-13</p> <p>GRAPHICON PICTURE DISK-1. REQUIRES PIXFILES/BAS FROM PD-12 & JOYSTICK</p> <p>PICTURES GCM 1 B 68</p> <hr/> <p>PD-14</p> <p>GRAPHICON PICTURE DISK-2. REQUIRES PIXFILES/BAS FROM PD-12 & JOYSTICK</p> <p>PICTURES GCM 1 B 68</p> <hr/> <p>PD-15</p> <p>GRAPHICON PICTURE DISK-3 REQUIRES PIXFILES/BAS FROM PD-12 & JOYSTICK</p> <p>PICTURES GCM 1 B 68</p> <hr/> <p>PD-16</p> <p>GRAPHICON PICTURE DISK-4 REQUIRES PIXFILES/BAS FROM PD-12 & JOYSTICK</p> <p>PICTURES GCM 1 B 68</p> <hr/> <p>PD-17 DISK UT.</p> <p>64KBHW BAS 0 A 1 AUTOSTRT BAS 0 B 1 BAKDIR BAS 0 A 3 BIN>BAS BAS 0 A 1 CASSLABL BAS 0 B 1 CURSOR BAS 0 B 1 CUSTOM BAS 0 B 3 CUSTOMIZ BAS 0 B 1 DIR BIN 2 B 1 DIR32 BAS 0 A 2 DIR32C DOC 1 A 3 DIRLISTR BAK 0 B 1 DIRLISTR BAS 0 B 1</p> <hr/> <p>PD-18 TAPE TO DISK DISK UTILITIES</p> <p>DIRSORT BAS 0 A 1 DISK-DIR BAS 0 A 1 DISKLABL BAS 0 A 1 LOADSOLU BAS 0 B 1 MENU BAS 0 B 1 PDIR BAS 0 A 1 SORT BAS 0 B 1 SORTPRT BAS 0 B 1 SORTSAVE BAS 0 A 1 SOULTION BIN 2 B 1 SUPERBAC BIN 2 B 1 T2D BIN 2 B 2 TIMER BAS 0 B 1 TPTODSK BIN 2 B 1</p> <hr/> <p>* PD-19 GAMES</p> <p>3DMAZE BAS 0 A 2 BOXES BAS 0 B 1</p>	<p>CLOSE EN BAS 0 B 2 CRITICAL BAS 0 B 1 GAMMON BAS 0 B 3 GOLDMINE BAS 0 A 3 HOCKEY BAS 0 A 1 HOGJOWL BAS 0 A 8 HORSERAC BAS 0 A 3 JUMPING BAS 0 B 1 KALIDESC BAS 0 B 1 MASTMIND BAS 0 B 1 MEMORY BAS 0 B 1 MOONBASE BAS 0 B 2 NAMES BAS 0 B 4 OHELLO BAS 0 B 4</p> <hr/> <p>* PD-20 GAMES</p> <p>PEG BAS 0 B 3 RABBIT BAS 0 B 1 SAFE BAS 0 B 2 SAUACER BAS 0 B 1 SHOOTEM BAS 0 B 2 SIMMON BAS 0 A 1 SLITHER BAS 0 A 2 SPACE WA BAS 0 B 4 STAR TRE BAS 0 B 1 SUBCHASE BAS 0 B 2 SUBDESTR BAS 0 B 2 SUNDANCE BAS 0 B 2 TANKS BAS 0 B 2 TOWER BAS 0 B 2 UNROVER BAS 0 B 1</p> <hr/> <p>PD-21 MUSIC</p> <p>PLAY MUSIC THROUGH YOUR TV OR MONITOR. COMPOSE, EDIT MUSIC.</p> <p>ORCH BIN 2 B 8 ORCH DOC 1 A 3 OCNVRT BIN 2 B 2 GHOSBUST MUS 4 M 3 STELMO MUS 4 M 2 MASH MUS 4 M 2 BOND1 MUS 4 M 2 2001 MUS 4 M 2 ARIA MUS 4 M 2 INVENTI MUS 4 M 1 BATTSTAR MUS 4 M 2 BOND2 MUS 4 M 2 CLOSENCT MUS 4 M 2 SCARBORO MUS 4 M 1 FUGUEINC MUS 4 M 1 MINUET MUS 4 M 1 LONGTIME MUS 4 M 2 MESSIAH MUS 4 M 3</p> <hr/> <p>* PD-22 MUSIC-1</p> <p>LOADM "NAME/MUS" EXEC TO PLAY MUSIC THROUGH TV OR MON.</p> <p>ADDPLAY BAS 0 B 1 DEPLAY BAS 0 B 1 MSQUEZ BAS 0 B 2 ALSOSPAK MUS 2 B 5 BOOGIE MUS 2 B 5 CIRCUS MUS 2 B 5 CLOWN MUS 2 B 2 CLOWNS MUS 2 B 4 HAYDEN MUS 2 B 8 JBGOOD MUS 2 B 4 PEACE MUS 2 B 2 PEACH MUS 2 B 5 PUFF MUS 2 B 6 GOODDIEY MUS 2 B 4</p>
---	---	--	--	---

* PD-23 MUSIC-2
 LOADM "NAME/MUS"
 EXEC TO PLAY MUSIC
 THROUGH TV OR MON.

ADDPLAY BAS 0 B 1
 DEPLAY BAS 0 B 1
 MSQUEZ BAS 0 B 2
 RAIN MUS 2 B 2
 SONATA3 MUS 2 B 3
 STRAV MUS 2 B 4
 FOGGY MUS 2 B 4
 FUNERAL MUS 2 B 3
 HARDDAY MUS 2 B 2
 INVENT MUS 2 B 2
 INVENT11 MUS 2 B 3
 INVENT15 MUS 2 B 3
 INVENT7 MUS 2 B 3
 INVENT8 MUS 2 B 2
 JOPLIN MUS 2 B 4
 KHAN MUS 2 B 6

* PD-24 MUSIC-3
 LOADM "NAME/MUS"
 EXEC TO PLAY MUSIC
 THROUGH TV OR MON.

ADDPLAY BAS 0 B 1
 DEPLAY BAS 0 B 1
 MSQUEZ BAS 0 B 2
 PEANUTS MUS 2 B 3
 ROCK MUS 2 B 5
 ROXANNE MUS 2 B 5
 SCHERZO MUS 2 B 2
 TEACH MUS 2 B 2
 PIANOMAN MUS 2 B 5
 STRANGER MUS 2 B 5
 CAMELOT MUS 2 B 4
 CHACONNE MUS 2 B 6
 DIAMOND MUS 2 B 3
 DOWNROAD MUS 2 B 4
 FANTASY1 MUS 2 B 2

* PD-25 MUSIC-4
 LOADM "NAME/MUS"
 EXEC TO PLAY MUSIC
 THROUGH TV OR MON.

FANTASY2 MUS 2 B 3
 GRENGRAS MUS 2 B 4
 HUMOR MUS 2 B 4
 INCROW MUS 2 B 3
 STARWARS MUS 2 B 2
 SUITEGM MUS 2 B 6
 SUPERMAN MUS 2 B 2
 WHENIM64 MUS 2 B 4
 ROOTBEER MUS 2 B 7
 WAYUARE MUS 2 B 3
 AXELF MUS 2 B 2
 TOCATA MUS 2 B 3

* PD-26 LAST WILL
 LOAN BAS 0 B 1
 LASTWILL BAS 0 B 6
 IMEGA BAS 0 B 3
 AWARI BAS 0 B 1
 BACARAT BAS 0 B 2
 BAGELS BAS 0 B 1
 BLACKJAC BAS 0 B 1
 CHUCK BAS 0 B 1
 CONCENTR BAS 0 B 1
 CUBES BAS 0 B 2

* PD-27 GAMES
 DEFUZE BAS 0 B 1
 DR ZEE BAS 0 B 1

FLIPFLOP BAS 0 B 1
 GO-FISH BAS 0 B 2
 HANGMAN BAS 0 B 2
 HIGHLOW BAS 0 B 1
 JACKPOT BAS 0 B 1
 KEYS BAS 0 B 1
 L E M BAS 0 B 3
 LUNARLD BAS 0 B 2
 NUMBERS BAS 0 B 1
 OBSTACLE BAS 0 B 1
 POOLGAME BAS 0 B 4
 RETURN BAS 0 B 1
 REVERSI BAS 0 B 2
 STARTREK BAS 0 B 2
 TTREK BAS 0 B 3

PD-28 COMM.CC-TALK
 BBS, TERM
 BBS'S DAT 1 A 1
 CCT IO 2 B 1
 CCTALK BAS 0 B 1
 CNFG40V1 BAS 0 A 5
 CNFG40V2 BAS 0 A 4
 CTLKEY BAS 1 A 1
 MTERM1 DOC 1 A 11
 MTERM2 DOC 1 A 8
 MTERM40 BIN 2 B 8
 REDIAL BAS 0 A 1
 PACREDIA BAS 0 A 1

PD-29 COMM, WORD
 PRO, GAMES
 GOSTSHIP BAS 0 B 8
 INT RATE BAS 0 B 2
 INVSTANL PC 0 B 4
 MENU BAS 0 B 4
 MOTOJUMP BAS 0 B 3
 SCREEN MAX 2 B 6
 SCREEN1 BIN 2 B 3
 SCREEN2 BIN 2 B 3
 SCREEN2 MAX 2 B 6
 STRINGTU BAS 0 B 4
 TTERM DSK 2 B 4
 TTHELP DAT 1 A 4
 USING BAS 0 B 3
 WF-DOC JP 0 B 2
 WORDFILE JP 0 B 4
 PARM1 DAT 1 A 1

PD-30 CHECK BOOK,
 UTILITIES
 CHECKBOK BAS 0 B 4
 CHECKBOK DOC 1 A 9
 DIRR CMD 2 B 1
 DVIEW BAS 0 B 1
 FILEMAID BAS 0 B 2
 LISTER BAS 0 B 1
 PAINTPOT BAS 0 B 4
 SCREEN MAX 2 B 6
 SCREEN1 BIN 2 B 3
 SCREEN2 BIN 2 B 3
 SCREEN2 MAX 2 B 6
 SPECZAP BAS 0 B 5
 TAPETYPE BIN 2 B 1
 TTERM DSK 2 B 4
 DVIEW DSK 0 B 1
 MENU BAS 0 B 4

PD-31
 PIRATES TREASURE.As
 you explore the cave
 looking for the
 treasure, a picture
 appears on the
 screen as you go
 from room to room.

These pictures are
 loaded from disk. A
 computer with disk
 drive is required.

PD-32
 Color Computer 3
 moving pictures.
 Consists of a
 beautiful waterfall
 and a colorful
 bouncing ball.

WATRFALL BAS 0 B 1
 WATRFALL BIN 2 B 1
 WATRFALL MGE 1 B
 BALL BAS 0 B 1
 BALL2 BAS 0 B 1
 BOUNCE BIN 2 B 1
 BALL2 HR1 2 B 4
 BALL2 HR2 2 B 4
 BALL2 HR3 2 B 4
 BALL2 HR4 2 B 4

* PD-33
 EDUCATIONAL PROGRAMS

ABBREV BAS 0 B 4
 ABCPOP BAS 0 B 3
 ALPHAAL BAS 0 B 1
 EDUCATE BAS 0 B 1
 HANGP BAS 0 B 1
 HOMONYM BAS 0 B 1
 SPELWORD BAS 0 B 1
 MATH BAS 0 B 2
 DRILL BAS 0 B 2
 MLTP BAS 0 B 1
 ROUND BAS 0 B 2
 AREA BAS 0 B 5
 METCONV BAS 0 B 3
 NUMBERS BAS 0 B 2
 SIEVE BAS 0 B 1

* PD 35
 ADDRESS FILES AND
 FINANCE PROGRAMS

PHONE BAS 0 B 1
 LABELPRT BAS 0 B 1
 LETTER BAS 0 B 3
 MAILST BAS 0 B 2
 PHONLST BAS 0 B 1
 MINIWORD BAS 0 B 2
 LNWIDTH BAS 0 B 1
 CHKWRITE BAS 0 B 2
 CHKANAL BAS 0 B 4
 PRNTCHK BAS 0 A 1
 CHECKS BAS 0 B 4
 CHCKSTUB BAS 0 B 1
 TOTALS DAT 1 A 1
 CHECKS DAT 1 A 1
 GRAPH BAS 0 B 4
 LOAN BAS 0 B 3
 CALC BAS 0 B 1
 PAYMENT BAS 0 B 1
 CASHJNL BAS 0 B 3
 AMORT BAS 0 B 3

* PD 36
 COMP.SCIENCE PGMS 1
 These programs are
 tutorials on basic
 programming.
 COMPSC1 BAS 0 B 8

COMPSC2 BAS 0 B 3
 COMPSC3 BAS 0 B 9
 COMPSC4 BAS 0 B 5
 COMPSC5 BAS 0 B 9
 COMPSC6 BAS 0 B 5
 GETPUT BAS 0 B 2

* PD 37
 COMP.SCIENCE PGMS 2

These programs are
 tutorials on basic
 programming.

IFTHEN BAS 0 B 9
 EXTENDED BAS 0 B 2
 GETPUT BAS 0 B 2
 COMPSC18 BAS 0 B 8
 COMPSC19 BAS 0 B 5
 COMPSCI7 BAS 0 B 7
 EXTDEMO BAS 0 B 3

* PD 38
 EDUCATIONAL PGMS
 These programs are
 excellent learning
 tools for school
 children.

ABBREV BAS 0 B 4
 ABCPOP BAS 0 B 3
 ALPHAAL BAS 0 B 1
 EDUCATE BAS 0 B 1
 HANGP BAS 0 B 1
 HOMONYM BAS 0 B 1
 SPELWORD BAS 0 B 2
 MATH BAS 0 B 2
 DRILL BAS 0 B 2
 MLTP BAS 0 B 1
 ROUND BAS 0 B 2
 AREA BAS 0 B 5
 METCONV BAS 0 B 3
 NUMBERS BAS 0 B 2

* PD 39
 ADDRESS FILES AND
 FINANCE PROGRAMS

PHONE BAS 0 B 1
 LABELPRT BAS 0 B 1
 LETTER BAS 0 B 3
 MAILST BAS 0 B 1
 WORDPROC BAS 0 B 3
 MAILST BAS 0 B 2
 PHONLST BAS 0 B 1
 MINIWORD BAS 0 B 2
 LNWIDTH BAS 0 B 1
 CHKWRITE BAS 0 B 2
 CHKANAL BAS 0 B 4
 PRNTCHK BAS 0 A 1
 CHECKS BAS 0 B 4
 CHCKSTUB BAS 0 B 1
 TOTALS DAT 1 A 1
 CHECKS DAT 1 A 1
 GRAPH BAS 0 B 4
 LOAN BAS 0 B 3
 CALC BAS 0 B 1
 PAYMENT BAS 0 B 1
 CASHJNL BAS 0 B 3
 AMORT BAS 0 B 3

PD-40
 TAPE-DSK & DSK-TAPE
 With these programs
 you can copy a disk
 to tape or a tape
 to disk.

T2D BIN 2 B 2
 DTCOPY BIN 2 B 1
 DSK-TP BAS 0 B 1
 DISKLIST BAS 0 B 1
 DIRLIST BAS 0 B 2
 DISKDUMP BAS 0 B 1
 CASSDIR BAS 0 B 1

* PD-41
 Picture files

STAMPS MAX 2 B 3
 STARTREK MAX 2 B 3
 SJ-TREK2 MAX 2 B 3
 SCHOOL MAX 2 B 3
 SATURN MAX 2 B 3
 ESCHER MAX 2 B 3
 LABOR MAX 2 B 3
 MASK MAX 2 B 3
 BUG BOX MAX 2 B 3
 SPACE MAX 2 B 3
 EASTER MAX 2 B 3
 SPACE 2 MAX 2 B 3
 POPEYE MAX 2 B 3
 GARFIELDS MAX 2 B 3
 BEETLE B MAX 2 B 3
 POLO MAX 2 B 3
 HAGAR MAX 2 B 3
 X-PAD MAX 2 B 3
 CASTLE MAX 2 B 3
 MUSIC TV MAX 2 B 3
 COCO MAX 2 B 3

* PD-42
 Picture files

TITLES MAX 2 B 3
 PIXFILES BAS 0 B 3
 THOLIAN MAX 2 B 3
 3001AD MAX 2 B 3
 F15 MAX 2 B 3
 QUEEN MAX 2 B 3
 BRONCOS MAX 2 B 3
 STARTREK MAX 2 B 3
 ROOM MAX 2 B 3
 RAMBO MAX 2 B 3
 OWL MAX 2 B 3
 ENTERPR MAX 2 B 3
 STAR-T3 MAX 2 B 3
 NCC-1701 MAX 2 B 3
 SAT-2 MAX 2 B 3
 ATMOSP MAX 2 B 3
 STARWARS MAX 2 B 3
 ORIENTAL MAX 2 B 3

* PD-43
 Picture files

STAMP MAX 2 B 3
 STRIPE MAX 2 B 3
 WOMAN MAX 2 B 3
 BLUEJAY MAX 2 B 3
 LUCY MAX 2 B 3
 OLD ENG MAX 2 B 3
 MENU1 MAX 2 B 3
 OWL MAX 2 B 3
 VAN GOG MAX 2 B 3
 WOMAN1 MAX 2 B 3
 PSH MAX 2 B 3
 DUCKPOND MAX 2 B 3
 RANGER MAX 2 B 3
 PLANET MAX 2 B 3
 CHRSTMAS MAX 2 B 3
 PEACE MAX 2 B 3
 WOMAN3 MAX 2 B 3
 HAWK MAX 2 B 3
 PHASER MAX 2 B 3
 PIXFILES BAS 0 B 3

```

PD-44      MAZE      BAS 0 B 4   PYTHON    BAS 0 B 2   HELLO     BIN 2 B 3   SUNSET    BAS 0 B 3
Terminal pgm with      DISKZAPP  BAS 0 B 2   LUNAR     BAS 0 B 2   GROVER    BIN 2 B 3   S NICKS   BAS 0 B 4
documentation. This    ZAP       BAS 0 B 3   LUNALANA  BAS 0 B 1   DRIVE IN  BIN 2 B 3   SNOOPY1   BAS 0 B 3
will work with the     DETHSHIP  BAS 0 B 3   AMAZING   BAS 0 B 2   TIME      BIN 2 B 3   MICKEY    BIN 1 B 8
CoCo-3. Instruc-      BACKUP35  BAS 0 B 1   BALOON    BAS 0 B 1   KOALA     BIN 2 B 3   DONALD    BIN 2 B 8
tions are included.    BOOT      BAS 0 B 1   VAPORWRM BAS 0 B 2   PATTERN   BIN 2 B 3   SNOOPY2   BAS 0 B 4
MTRM43     BIN 2 B 8   DOSSTART  BAS 0 B 1   BULLSEYE  BAS 0 B 1   HAGAR    BIN 2 B 3   SNOOPY3   BAS 0 B 4
CONFIG43   BAS 0 B 4   LABEL     BAS 0 B 2   CRASH     BAS 0 B 1   CHIPS     BIN 2 B 3   SNOOPY4   BAS 0 B 4
MTSTART    BAS 0 B 4   DSKDSABL BAS 0 B 1   DOTS      BAS 0 B 3
MTERM1     DOC 1 A 11  NOFREEG   BAS 0 B 1   F-16      BAS 0 B 3
MTERM2     DOC 1 A 8   FORMATER  BAS 0 B 1   KRYPTON   ART 2 B 3
MTERM3     DOC 1 A 7   ROMRAM    BIN 2 B 1   KRYPTON   BAS 0 B 1
DOS BOOT   DAT 1 A 1   SUPDUP    BIN 2 B 1   KRYPTON   GAM 0 B 1
"          " 0 B 1   TESTTEXT  BAS 0 B 1
"          " 1 A 1
READD0C    BAS 0 B 1
-----
* PD-45
Picture Files
DRAGON     MAX 2 B 3
HOT LIPS   MAX 2 B 3
ANIMALS    MAX 2 B 3
CLOWN F    MAX 2 B 3
FISH       MAX 2 B 3
3 MEN      MAX 2 B 3
S MAP      MAX 2 B 3
BUGS       MAX 2 B 3
CFISH      MAX 2 B 3
HERO       MAX 2 B 3
WMAP       MAX 2 B 3
GSCOTT     MAX 2 B 3
STATES     MAX 2 B 3
HORSE      MAX 2 B 3
CROSS      MAX 2 B 3
FOODW      MAX 2 B 3
RSTONE     MAX 2 B 3
COCO       MAX 2 B 3
ALIEN      MAX 2 B 3
PIXFILES   BAS 0 B 3
-----
* PD-46
Talk and Music
Files (C)LOADM
"FILE" then EXEC.
TALK       BIN 2 B 11
TALK2      BIN 2 B 11
WILLTELL   BIN 2 B 9
MUSICBOX   BIN 2 B 1
BEATLES    BIN 2 B 4
JUMP       BIN 2 B 5
GRELN      BIN 2 B 5
GHOST      BIN 2 B 4
JINGLE      BIN 2 B 3
WORLD      BIN 2 B 5
CRYROAD    BIN 2 B 2
-----
* PD-47
Miscellaneous Pgms
T          BAS 0 B 2
SANTEE2    BAS 0 B 1
MILEAGE    BAS 0 B 1
M          BAS 0 B 1
DIGITS     BAS 0 B 1
NUMBLIST   BAS 0 B 1
COUNT     BAS 0 B 1
SC         BAS 0 B 1
DRAWTEXT   BAS 0 B 1
SAMPLE     BAS 0 B 1
GRSCRWRT   BAS 0 B 2
HRTEXT2    BAS 0 B 3
DRAW       BAS 0 B 2
WRITER     BAS 0 B 1
TYPEBET    BAS 0 B 2
WRITEBET   BAS 0 B 2
TEXT2      BAS 0 B 2
SANTEE     BAS 0 B 2
SHUTTLE    BAS 0 B 1
AJOCK      BAS 0 B 1
PLATFORM   BAS 0 B 1
-----
* PD-48
Miscellaneous Pgms
EXTBAS     BAS 0 B 3
DISAPPEAR  BAS 0 B 1
PAINT      BAS 0 B 1
DATA       BIN 2 B 1
DATA2      BIN 2 B 1
SCRDATA    BIN 2 B 1
FILL2      BIN 2 B 2
QUADDRAW   BAS 0 B 1
CELTIC     BAS 0 B 2
ALL RAM    BAS 0 B 1
CHARGEN    BIN 2 B 1
ROMRAM     BIN 2 B 1
OBSTACLE   BAS 0 B 1
64K RAM    BAS 0 B 1
COLORSEL   BAS 0 B 1
TRIG       BAS 0 B 4
ALGEBRA    BAS 0 B 4
PLAY       BAS 0 B 1
STATECAP   BAS 0 B 2
MLSOUNDS   BAS 0 B 1
ROTATION   BAS 0 B 2
PARABOLA   BAS 0 B 2
INSTAPIC   BAS 0 B 1
CLOVER     BAS 0 B 1
HAT-PLOT   BAS 0 B 1
WHEEL 1    BAS 0 B 1
LETTER-R   PAR 1 A 1
3-LINES    ROT 1 A 1
TRAPZOID   ROT 1 A 2
PYRAMID    ROT 1 A 2
CUBE       ROT 1 A 3
51X24      BAS 0 B 2
WINDOW     BAS 0 B 5
GGPRISU    BAS 0 B 1
KALEIDO    BAS 0 B 1
OKB3APRT   BAS 0 B 1
NUMCNVTR   BAS 0 B 1
ADVRTN     BAS 0 B 1
-----
* PD-49
Miscellaneous Pgms.
BC         BIN 2 B 10
PEDRO     BIN 2 B 11
BLOCKADE   BAS 0 B 3
REPEAT     BAS 0 B 1
AIRPLANE   BAS 0 B 1
BUSTOUT    BAS 0 B 1
GOLF       BAS 0 B 7
CITY       BAS 0 B 2
AIR-RAID   BAS 0 B 2
MAZE       BAS 0 B 4
DUALDUP    BIN 2 B 2
DIRMAP     BAS 0 B 3
CHESS      BAS 0 B 5
WHATZIT    BAS 0 B 4
BATLSHIP   BAS 0 B 3
SP*ROCKS   BAS 0 B 1
-----
* PD-50
Miscellaneous PGMS
GOBBLER    BAS 0 B 2
-----
* PD-51
Games & Programs
DRAGRACE   BAS 0 B 1
WORMER     BAS 0 B 2
SIMON      BAS 0 B 2
RIDER      BAS 0 B 2
MISSILE    BAS 0 B 3
LETSHOOT   BAS 0 B 2
SHOOTGAL   BAS 0 B 2
MISSILE2   BAS 0 B 3
FENCE      BAS 0 B 3
BANDIT     BAS 0 B 1
CHICKEN    BAS 0 B 2
MAXIMUM    BAS 0 B 3
FLIGHT     BAS 0 B 2
COVERUP     BAS 0 B 2
WORLDMAP   BAS 0 B 4
POUNCE     BAS 0 B 1
MARTIANS   BAS 0 B 2
FINDIT     BAS 0 B 3
SCRAMBLE   BAS 0 B 5
BOUNBABY   BAS 0 B 2
CHICK      BAS 0 B 3
BOBO       BAS 0 B 3
RUBIC      BAS 0 B 4
MCJUMP     BAS 0 B 3
-----
* PD-52
Picture files
COCO       MAX 2 B 6
COL COCO   MAX 2 B 6
MOOSHEAD   MAX 2 B 6
COKE       MAX 2 B 6
CUBS       MAX 2 B 6
REDS       MAX 2 B 6
BREAKERS   MAX 2 B 6
USFL       MAX 2 B 6
SPACE      BIN 2 B 3
GIZMO      MAX 2 B 3
DINASOUR   MAX 2 B 3
-----
* PD-53
Picture Files
INDIAN     MAX 2 B 6
HOMECOME   MAX 2 B 6
GRIN       BIN 2 B 3
TARD       BIN 2 B 3
STUD       BIN 2 B 3
COMET      BIN 2 B 3
DESERT     BIN 2 B 3
FOOD       BIN 2 B 3
SMIRK      BIN 2 B 3
PLAYA      BIN 2 B 3
-----
* PD-54
Picture Files
PENTAGON   PIC 2 B 3
GRID 2     PIC 2 B 3
SNOWFLAK   PIC 2 B 3
CONETUNL   PIC 2 B 3
4-POINT    PIC 2 B 3
BALSTR     MAX 2 B 3
CARTOON    MAX 2 B 3
HUELEWIS   MAX 2 B 3
STARTREK   MAX 2 B 3
HOUSE1     MAX 2 B 6
HOUSE2     MAX 2 B 6
LIFECYCL   MAX 2 B 6
COCOMAG    MAX 2 B 3
MASCATL    MAX 2 B 3
COLUMBIA   MAX 2 B 3
POLO       MAX 2 B 3
ET         BAS 0 B 7
WHEEL 1    PIC 2 B 3
-----
* PD-55
Picture Files
PARKERPT   MAX 2 B 3
TOWER      PIC 2 B 3
TOWER2     PIC 2 B 3
SCREEN      PIC 2 B 3
BOMB       PIC 2 B 3
ANDRON     PIC 2 B 3
SALE       PIC 2 B 3
CHIPS      PIC 2 B 3
TUNLROAD   BIN 2 B 3
LONEROAD   BIN 2 B 3
CITYROAD   BIN 2 B 3
LAKEROAD   BIN 2 B 3
CROSSROAD  BIN 2 B 3
BLACK      BIN 2 B 3
CAL1       BIN 2 B 3
CAL2       BIN 2 B 3
CAL3       BIN 2 B 3
3-LEAF     PIC 2 B 3
5-STARS    PIC 2 B 3
SPHERE     PIC 2 B 3
15-LEAF    PIC 2 B 3
-----
* PD-56
Glossary, Memory
Maps, Programs
COCO       VIP 1 A 4
VIP ON 3   VIP 1 A 1
BEEF       VIP 1 A 1
MCTRM3     VIP 1 A 1
GLOSSARY   VIP 1 A 7
POKEPEEK   VIP 1 A 17
WIDTH      VIP 1 A 1
COCO 3     VIP 1 A 17
MISSLES    BAS 0 B 2
CLOCK      BAS 0 B 1
JET        BAS 0 B 4
-----
* PD-57
Picture Files
VAMPIRE    PIC 2 B 3
ATLANTA    BAS 0 B 3
NOGHOST    PIC 2 B 3
AIRPORT    BAS 0 B 4
S EASTON   BAS 0 B 4
ISMLSTEP   BAS 0 B 4
HAGAR      PIC 2 B 3
-----
* PD-58
Miscellaneous Pgms
DISKLIST   BAS 0 B 1
DIRLIST    BAS 0 B 2
ML ADDR    BAS 0 B 1
DISKDUMP   BAS 0 B 1
PRINUTIL   BAS 0 B 2
CALPRINT   BAS 0 B 3
ALPHSONG   BAS 0 B 1
PAINT      BAS 0 B 1
DOGPIC     BAS 0 B 2
EVADER     BAS 0 B 1
NUKATTC    BAS 0 B 2
BASICMAP   BAS 0 B 3
JOYPAINT   BAS 0 B 1
PUMPKIN    BAS 0 B 1
HOMOYMS    BAS 0 B 1
ABBREV     BAS 0 B 4
CONVERT    BAS 0 B 3
CASDIR     BAS 0 B 1
CVERT      BAS 0 B 1
FLASCARD   BAS 0 B 1
MESSAGE    BAS 0 B 1
RELOCAT    BAS 0 B 1
COUNT     BAS 0 B 1
CALENDAR   BAS 0 B 1
DOGS       BAS 0 B 1
DOGFIGHR   BAS 0 B 1
BEAST      BAS 0 B 1
-----
* PD-59
GAMES, UTILITIES
64X64F     BAS 0 B 1
RND#'S     BAS 0 B 1
SCROLLER   BAS 0 B 1
COCOBUG    BAS 0 B 2
DRWBOARD   BAS 0 B 1
SPACE      BAS 0 B 1
DIR-ADDR   BAS 0 B 1
BACKGAMN   BIN 2 B 2
CHESS      BIN 2 B 3
BATTLE     BIN 2 B 2
GERM       BIN 2 B 1
BLEEP      BAS 0 B 2
TICKER     BAS 0 B 3
LEAKYTAP   BAS 0 B 3
UTOPIAN    BAS 0 B 4
COLORDOT   BAS 0 B 3
STAYALIV   BAS 0 B 2
TIMEFLT    BAS 0 B 3
NAVYGUNS   BAS 0 B 2
ATACMAN    BAS 0 B 3
CALENDAR   BAS 0 B 1
POKER25    BAS 0 B 1
VIEWERS    BAS 0 B 1
STUFF      BAS 0 B 1
-----
* PD 60 Basic Pgms
S NICKS    BAS 0 B 4
1SHLSTEP   BAS 0 B 4
SUNSET     BAS 0 B 3
3DITT      BAS 0 B 4
BATTSHIP   BAS 0 B 2
CRACE      BAS 0 B 2
FLY        BAS 0 B 3
KINGS      BAS 0 B 6
KINGTUT    BAS 0 B 7
OREGON     BAS 0 B 9
POKER      BAS 0 B 2
ROBOTS     BAS 0 B 3
ROLLON     BAS 0 B 2
SORCERER   BAS 0 B 6

```

* PD-61 Pictures		SLOTS	BAS	0	B	2	CASSDIR	BAS	0	B	1	PD-69 Disk Utilities	JETI	BIN	2	B	3								
HAGAR	PIC	2	B	3	-----	-----	CONTOUR	BAS	0	B	1	DIRPER	BAS	0	B	2	MOONLIT	BAS	2	B	3				
SHIPS	BAS	0	B	2	-----	-----	CONVERGE	BAS	0	B	1	DISKLOOK	BAS	0	B	1	RONDO	BIN	2	B	4				
SIGNS	BAS	0	B	1	* PD-64 Basic Pgms	-----	CONVERT	BAS	0	B	3	DKTODK	BAS	0	B	3	LOOKLOVE	BIN	2	B	1				
SPACE	BAS	0	B	8	-----	-----	COUNT	BAS	0	B	1	DSK2TP	BAS	0	B	2	MENU	BAS	0	B	1				
3GUYS	MAX	2	B	3	OHMSLAW	BAS	0	B	1	DEC()HEX	BAS	0	B	1	DSKLIERY	BAS	0	B	3	WE	BAS	0	B	1	
AIRPORT	BIN	2	B	6	POWER UP	BAS	0	B	1	FUELCOST	BAS	0	B	1	DSKMSTER	BAS	0	B	4	MESSAGE	BAS	0	B	1	
BIGCAT	MAX	2	B	3	ROMPACK	BAS	0	B	1	HEXLOAD	BAS	0	B	1	DSKSPEED	BAS	0	B	1	OBJECTS	BIN	2	B	2	
CUBE	BIN	2	B	3	ROMRAM	BAS	0	B	1	HEXTODEC	BAS	0	B	1	DTOD	BAS	0	B	2	ODIE	PIX	2	B	3	
DOGPICT	BAS	0	B	2	SCRDUMP	BAS	0	B	1	IN-OUT	BAS	0	B	1	D UTIL	BAS	0	B	2	TANK	BIN	2	B	4	
EARTH	MAX	2	B	3	SLOSKROL	BAS	0	B	1	HOMONYMS	BAS	0	B	1	DIRGET	BAS	0	B	1	TRIANGLE	BIN	2	B	2	
GARFIELD	PIX	2	B	3	SORT	BAS	0	B	1	JOYPAINT	BAS	0	B	1	DIRSAVE	BAS	0	B	1	WORLDMAP	BIN	2	B	4	
GIRL	MAX	2	B	3	SPEDMATH	BAS	0	B	3	KALVOS	BAS	0	B	1	DISK FIX	BAS	0	B	1	PAINT	BAS	0	B	1	
NEUWAVE	MAX	2	B	3	SPOOLER	BIN	2	B	1	LINES	BAS	0	B	1	DISKDIRE	BAS	0	B	3	SCRDATA	BIN	2	B	1	
OLIVER	MAX	2	B	3	UPPER32K	BAS	0	B	1	MACDATA	BAS	0	B	1	DISKDUMP	BAS	0	B	1	RES	BAS	0	B	1	
OWL	MAX	2	B	3	STRIKE	BAS	0	B	1	MISSLETT	BAS	0	B	1	DISKEDIT	BAS	0	B	4	SCAN	BAS	0	B	1	
PEANUTS	PIX	2	B	3	SHIPS	BAS	0	B	2	-----	-----	DISKLIST	BAS	0	B	1	* PD 72 Basic and								
SHUTTLE	MAX	2	B	3	WILLSADV	BAS	0	B	5	* PD-67 Basic Pgms		DISKSORT	BAS	0	B	1	Machine Lang. Pgms								
SR-71	MAX	2	B	3	RACEWAY	BAS	0	B	4	LOAN	BAS	0	B	3	DISKTEST	BAS	0	B	1	FIND	BAS	0	B	1	
ZEBCHES	MAX	2	B	3	TREK	BAS	0	B	4	LOANAMOR	BAS	0	B	1	DISKTIME	BAS	0	B	1	LOC FIND	BAS	0	B	1	
ZIGGY	PIX	2	B	3	TXTCNVRT	BAS	0	B	1	64KLOOK	BAS	0	B	8	DISKCLEAN	BAS	0	B	1	ML ADDR	BAS	0	B	1	
										ASSEMBLR	BAS	0	B	3	MASTRDSK	BAS	0	B	4	MLFINDER	BAS	0	B	1	
										DISASSY	BAS	0	B	4	-----	-----									
										FINANCE	BAS	0	B	8	* PD-70 Basic Pgms										
										ROMDUMP	BAS	0	B	1	MLADFND	BAS	0	B	2	READBIN	BAS	0	B	1	
										WEREWAND	BAS	0	B	5	BIGHILL	BAS	0	B	1	RELOCAT	BAS	0	B	1	
										CHECKS	BAS	0	B	4	BLACKJK	BAS	0	B	4	CHKBOOK	BAS	0	B	3	
										MONEYHLP	BAS	0	B	4	CIA	BAS	0	B	6	FINANAD	BAS	0	B	6	
										CHKBOOK	BAS	0	B	3	CIPHER	BAS	0	B	1	GRAPHICS	BAS	0	B	5	
										STAT-LOG	BAS	0	B	3	CUBES	BAS	0	B	1	HOMEUTIL	BAS	0	B	6	
										WORDPRC	BAS	0	B	5	DOG FIGHT	BAS	0	B	1	LIFE	BAS	0	B	4	
										WORDSCAR	BAS	0	B	2	FISH	BAS	0	B	1	MCONVERT	BAS	0	B	2	
										TYPING	BAS	0	B	2	FLIP	BAS	0	B	2	METCONV	BAS	0	B	1	
										-----	-----	FOOTBALL	BAS	0	B	4	JOYLIST	BAS	0	B	1	CLOCK	BIN	2	B
										* PD-68 Basic Pgms		GOLDMINE	BAS	0	B	3	CAMELOT	BIN	2	B	2	FIRE	BIN	2	B
										ART	BAS	0	B	1	HANGMAN	BAS	0	B	2	CLOCK	DAT	1	A	1	
										BAROGRAPH	BAS	0	B	1	HILOW	BAS	0	B	3	-----	-----				
										BEGIN	BAS	0	B	1	HOBBIT	BAS	0	B	2	* PD 73 Basic Pgms					
										BWDUMP	BIN	2	B	1	HUSTLE	BAS	0	B	1	CARTEL	BAS	0	B	7	
										CHAR	BAS	0	B	2	JUMP	BAS	0	B	1	DODGE-EM	BAS	0	B	2	
										COM	BAS	0	B	2	MEMORY	BAS	0	B	2	DOGS	BAS	0	B	1	
										DISMON	BAS	0	B	7	PROTECT	BAS	0	B	2	DOORS	BAS	0	B	1	
										DOT	BAS	0	B	1	QUEST	BAS	0	B	4	PINGPONG	BAS	0	B	1	
										EDITOR	BAS	0	B	3	SLITHER	BAS	0	B	1	CACAPHON	BAS	0	B	1	
										EXTNDKYB	BAS	0	B	4	STOCK	BAS	0	B	3	SUB	BAS	0	B	5	
										EXTNDKYB	DOC	1	A	7	-----	-----									
										FREE	BAS	0	B	1	* PD-71 Basic &					TREK	BAS	0	B	5	
										GRADBOOK	BAS	0	B	1	Machine Lang. Pgms					TYCOON	BAS	0	B	2	
										GRNDSTIFF	BAS	0	B	1	DISASSEM	BAS	0	B	2	SCRAMBLE	BAS	0	B	5	
										INSTR	BAS	0	B	1	PAYMENT	BAS	0	B	1	SIMON	BAS	0	B	2	
										LET	BAS	0	B	3	STATCAP	BAS	0	B	2	WHERISIT	BAS	0	B	2	
										STOCKS	BAS	0	B	5	TEMPCONV	BAS	0	B	1	WALLHIT	BAS	0	B	1	
										THOLINER	BAS	0	B	1	ECHOSONG	BAS	0	B	1	TICTACT	BAS	0	B	2	
										ATOMS	BAS	0	B	2	MUSCONV	BAS	0	B	1	CHBASIC	BAS	0	B	1	
										BEAST	BAS	0	B	1	FUGUE	BIN	2	B	3						

PUBLIC DOMAIN SOFTWARE

This large collection of programs will allow you to quickly expand your library. All programs are available on disk and programs with a * are available on tape. Some programs require a joystick. Instructions are included in some collections as DAT, DOC, or TXT files. Prices are as follows:

1-4 \$4.95, 5-9 \$4.50, 10-24 \$4.00, 25 up \$3.50

Add \$1 shipping for less than 10 and \$2 for 10 up.
Checks, VISA, or Master Cards.

Dynamic Electronics Inc.
P. O. Box 896 (205) 773-2758
Hartselle, AL 35640

C PROGRAMMING

NEW SERIES

by
Norm Matice

January always brings about thoughts of New Year's Resolutions. This year a good resolution may be to learn a new computer language. How about C? C is quite popular on mainframe systems especially those operating with Unix. Since OS-9 and Unix share many of the same characteristics C should be a natural for an OS-9 based Color Computer.

One of the nice things about C is that it doesn't change much from one computer to another. I have programmed C on an H-P minicomputer, an Intel mainframe, IBM-PC compatibles and of course the Color Computer. The language is the same from one to the next. If you write a program for one in C, the same program can be typed into one of the others, be compiled and executed.

If we decide to learn C, what will we need? We will need a Color Computer I, II or III, OS-9 level I or II, and 2 disk drives. On top of that we will need Radio Shack's C-Compiler for the CoCo. Not exactly a cheap beginning.

The first thing to do once we have gathered all our equipment in one place is make a backup of both your C compiler disk and your C library disk. Put the originals away for safe keeping. One other thing needs to be done to the compiler disk. It needs an editor. There is nothing wrong with using the text editor that comes with OS-9. It's handy and it's already paid for and we should be somewhat familiar with it. Put the OS-9 system disk in drive 0 and the

backup compiler disk in drive 1 and type in the following command.

```
OS9:
  copy /d0/cmds/edit /d1/
      cmds/edit
```

Now insert the backup compiler disk in drive 0 and the backup library disk in drive 1. At the OS-9 prompt type in the following:

```
OS9:chx /d0/cmds
OS9:chd /d1/sources
```

Whether you use upper or lower case is not critical, but the proper use of the space is. There must be a space between chx and /d0/cmds and one between chd and /d1/sources. At this point we would be ready to try and write a C program. The first rule to remember about C is that when you send a file to the C compiler to be compiled it must have a .c extension on it. If the C compiler doesn't see the format filename.c it will refuse to compile the program. Should you forget in naming your file to use the .c extension don't worry you have the OS-9 rename command.

For our first program lets try a little input/output of text with C. We will call the program file greet.c remembering to use the .c extension. Type in this line at the OS-9 prompt:

```
edit greet.c
```

This should get you into the text editor you transferred onto the compiler disk. Now type in

the following listing, taking care to duplicate the spacing of the first two lines of the program. Also remember when using the OS-9 editor that you must make the first character of a line a space, otherwise the editor will think you are giving it a command. The spacing of the other lines is not critical. Also whether you use upper or lower case letters is not critical on the Color Computer.

The braces on the line after main() and on the last line can be created by holding the clear key and pressing the < or > keys, on a level I system and the ctrl and < or > keys on a level II system. Which way the braces are facing is important. To get the brackets in the line char name[25]; hold the clear or ctrl key, depending on level, and press the (or) keys. Again the direction the brackets are facing is important. The slash such as the one in the first printf lines can be had by holding the clear or ctrl key, again level dependent, and pressing the backslash key.

That should about take care of the special characters. Now type in the listing below, being careful to include all the punctuation you see in the program. The semicolons and commas are a necessary part of the program.

```
#include <stdio.h>
main()

/* A simple computer greeting
program. */

char name[25];

printf("What is your name? n");
scanf("%s", name);
printf("Hello %s it is nice to
meet you. n", name);
```

Lets go over each line and try to see just what is going on here. The first line #include <stdio.h> includes the routines for standard input and output in

your program. While this is not needed in every program you write it's, a good habit to put it them in all of your programs anyway.

The second line main() will be found in every C program you will ever write or see. It tells the compiler that this is the main part of the program (we'll discuss subprograms at a later date), and the compiler won't compile without it. The left brace tells the compiler where the program starts. Its counterpart the right brace signifies the end of the program. They will also be used later to enclose parts of a program.

The line /* A simple computer greeting program. */ is just a comment. The next line (the blank line is optional) char name[25]; defines the variable name as a character and the [25] specifies the length of the character string. All variables must be defined in C. Also note the ; at the end of the line. All statements in C must end with a semi-colon, the compiler needs them.

The next line printf("What is your name? n"); is the equivalent of a BASIC print statement. Don't forget the f when doing the printf line. the n in the string tells the compiler to perform a carriage return. Without it the next line to be printed would appear on the same line as this one. As far as C is concerned the n is one character, so it is important that both halves be present.

The following line scanf("%s", name); is the same as a BASIC input line. The computer will wait for a response from you at this point before continuing. The %s tells the compiler that the response will be a string. Even if you type in a number it will be treated like a string. The second printf line, printf("Hello %s it is nice to meet you. n", name); simply prints the computer's

response after you tell it your name. The %s in this case holds open a space for the string name in the line. If you were to eliminate the spaces before and after the %s in the line, then the print out would look like this ... Hellownameit is....., with out the periods of course.

The last line is the right brace to show where the program ends. Now that we have the program in a file we are ready to compile it. Once out of the editor and back at the OS-9 prompt type the following.

```
cc1 greet.c
```

The compiler will start to work on the program. If all goes well a level I system will take about 7 minutes to compile and a level II system will take about 3.5 minutes. If things do not go well check to make sure your file is exactly like the example in punctuation and spacing. Now that your file is compiled you can run it. To do this type the following at the OS-9 prompt.

```
greet
```

The disk drive should start up, and after the program is loaded it will execute. The file greet will not be a text file. If you want to make changes to the program you will have to go back to the file greet.c. The greet.c file will remain on your /d1/sources directory. Greet will be in you /d0/cmds directory. Also greet is a machine language program now, this means you can move it to any disk and it will still execute when called on or it can be loaded into memory and executed from there, just like any OS-9 module can.

That's quite a bit to get started with. Try the program, get used to your system and next time we'll try another example.

```
EARLY TO BED
AND EARLY TO RISE
MAKES A MAN
HEALTHY, WEALTHY AND WISE
```

New Products

This section is available free to all producers of color computer products.

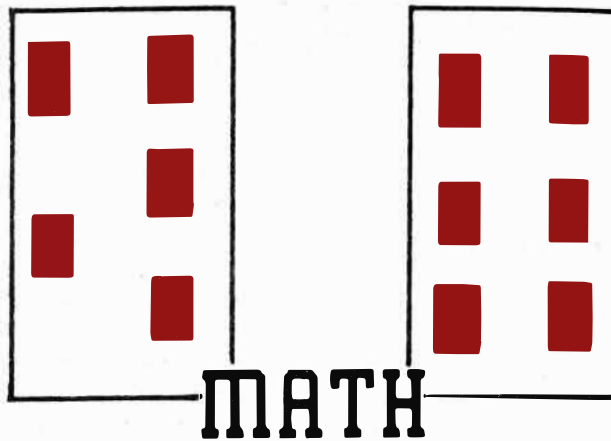
3 NEW SPORTSware products.

The first is their CATALOG ON DISK. This flippy disk has files on both sides (mostly graphic pages) which illustrate some of our most popular products. This catalog sells for \$3.00. However, if the owner buys anything from the catalog, the \$3.00 is deducted from the product cost. So, it's FREE if a purchase is made. Their CATALOG ON DISK runs only on the COCO 3 but contains some COCO 2 software also. On disk only.

Second is Joseph Paravati's COCO 3 WHEEL OF FORTUNE. This popular TV game show takeoff was first written for the COCO 2 but recently upgraded to take advantage of the COCO 3 graphics. It's still available for the COCO 2. The retail price is \$21.00. System requirements are a COCO 3 128K 1 disk drive.

Third is another new game by Paravati, BLACK GRID. This is a computer adaptation of PARKER BROS. BLACK BOX game. The computer hides from 2 to 9 blocks inside the black grid. The player must locate them by shooting "rays" into the grid. After each shot graphic symbols appear on the outer edges of the grid representing what the rays hit, if anything. It's quite a puzzle. Three different play modes are available. Retail is \$21.00. Requires a COCO 3 128K disk or tape.

*SPORTSware 1251 S. Reynolds
Suite 414 Toledo, OH 43615 (419)
389-1515.*



Try your addition and multiplication ability out with this game. It is a one player game that gives you the choice of adding or multiplying. Two cards are displayed on the screen with blocks on them. If you chose add, then you add the blocks for the answer. If you chose multiply, then you multiply the blocks. The computer keeps the score along with the number of problems you want. This program is provided by Bill Bernico Software and is used by permission.

```

10 'MATH DICE by Bill Bernico
    (C) 1988 BILL BERNICO SOFTWARE
20 CLS:PRINT@99,"ADDITION OR MULTIPLICATION~":PRINT@140,"(A / M)
30 C$=INKEY$:IF C$=""THEN30
40 IF C$="A"THEN D$="+~":B$="ADD DICE AND INPUT TOTAL~":GOTO70
50 IF C$="M"THEN D$="X~":B$="MULTIPLY DICE AND INPUT TOTAL~":GOTO70
60 GOTO 20
70 PRINT@326,"NUMBER OF PROBLEMS ~":INPUT
    C
80 CLS:FOR H=1 TO C:PRINT@0,B$
90 E$=CHR$(128):D=RND(6):PRINT@68,STRING$(9,207)::PRINT@100,STRING$(9,207)::PRINT@132,STRING$(9,207)::PRINT@164,STRING$(9,207)::PRINT@196,STRING$(9,207)::PRINT@228,STRING$(9,207)::PRINT@260,STRING$(9,207);
100 PRINT@82,STRING$(9,207)::PRINT@114,STRING$(9,207)::PRINT@146,STRING$(9,207)::PRINT@178,STRING$(9,207)::PRINT@210,STRING$(9,207)::PRINT@242,STRING$(9,207)::PRINT@274,STRING$(9,207);
110 PRINT@175,D$::ON D GOSUB 230,240,250,260,270,280
120 E=RND(6):ON E GOSUB 290,300,310,320,330,340

```

```

130 IF C$="A"THEN I=A+B
140 IF C$="M"THEN I=A*B
150 GOSUB350:PRINT@352,STRING$(2,143):PRINT@362,"~":INPUT"TOTAL~":J
160 IF J<>I THEN PRINT@352,STRING$(22,143):PRINT@360,"wrong try again~":POKE1389,45:POKE1393,32:FORX=1TO1000:NEXT:PLAY"O4T6OBAGFEDC~":PRINT@328,"~":G=G+1:GOSUB350:GOTO130
170 PRINT@352,STRING$(22,143):PRINT@364,"correct~":FORX=1TO1000:NEXT:PLAY"O5T6OCDEFGAB~":F=F+1:GOSUB 350:NEXT H
180 PRINT@328,"TRY AGAIN (Y/N)
190 A$=INKEY$:IF A$=""THEN190
200 IF A$="Y"THEN RUN
210 IF A$="N"THEN CLS:END
220 GOTO 190
230 PRINT@168,E$::A=1:RETURN
240 PRINT@101,E$::PRINT@235,E$::A=2:RETURN
250 PRINT@101,E$::PRINT@168,E$::PRINT@235,E$::A=3:RETURN
260 PRINT@101,E$::PRINT@107,E$::PRINT@229,E$::PRINT@235,E$::A=4:RETURN
270 PRINT@101,E$::PRINT@107,E$::PRINT@168,E$::PRINT@229,E$::PRINT@235,E$::A=5:RETURN
280 PRINT@101,E$::PRINT@107,E$::PRINT@165,E$::PRINT@171,E$::PRINT@229,E$::PRINT@235,E$::A=6:RETURN
290 PRINT@182,E$::B=1:RETURN
300 PRINT@115,E$::PRINT@249,E$::B=2:RETURN
310 PRINT@115,E$::PRINT@182,E$::PRINT@249,E$::B=3:RETURN
320 PRINT@115,E$::PRINT@121,E$::PRINT@243,E$::PRINT@249,E$::B=4:RETURN
330 PRINT@115,E$::PRINT@121,E$::PRINT@182,E$::PRINT@243,E$::PRINT@249,E$::B=5:RETURN
340 PRINT@115,E$::PRINT@121,E$::PRINT@179,E$::PRINT@185,E$::PRINT@243,E$::PRINT@249,E$::B=6:RETURN
350 PRINT@420,"RIGHT ANSWERS:~":F:PRINT@452,"WRONG ANSWERS:~":G:PRINT@484,"# OF PROBLEMS:~":C::RETURN

```

PRODUCT REVIEWS

A MAZING WORLD OF MALCOM MORTAR

for the Color Computer 3

Reviewed by Norm Matice

A Mazing World of Malcom Mortar is an arcade type game on ROM pack for the Color Computer III. As the title suggest the game has something to do with mazes. In the game you are B. Rick an apprentice bricklayer.

According to the introduction B. Rick winds up in the maze due to an on the job accident. Instead of getting workman's compensation B. Rick must battle his way through the mazes and over come the evil Malcom Mortar. The way he does that is by picking up bricks and dynamite.

There are two types of bricks, regular grey bricks and gold bricks. The grey bricks are used to build walls to trap fuzzies and boreheads in the maze. A fuzzie is a poison quill shooting creature that can harm B. Rick. a borehead is a mechanical minion that can drill through temporary walls. The temporary walls that B. Rick lays can be torn down with dynamite or by a borehead. Then can be made permanent by pressing the F2 key on the keyboard.

The gold bricks are used to imprision Malcom Mortar, whose very touch is fatal to B. Rick. Trapping Malcom Mortar is not an easy task. If you are successful though, B. Rick will be allowed to leave the maze.

So far I have been unable to get B. Rick out. Although the game sounds simple enough it does provide quite a challenge. My only complaint is sometimes B. Rick goes beyond where you want to construct a wall and the fast reaction of the joystick makes it hard to position him

back where you need him. Small movements of the character are hard to do.

The game is fast paced and the graphics and color scheme blend very well together. It is a very eye appealing game. It can be played on 128K CoCo III and the use of joysticks is optional, but suggested. A Mazing World of Malcom Mortar is available at Radio Shack stores and cost \$29.95.

ROGUE

Reviewed by Norm Matice

Color Computer 3 Game

Rogue is a Fantasy Role Playing (FRP) game. In the game you are the Rogue. Your job is to fight your way through the Dungeons of Doom, get the fabled Amulet of Yendor and get back out. While you are doing this you also try to stay alive.

The Dungeons of Doom consist of a multi-layer series of rooms. Each layer has 7 to 10 rooms on it. In the rooms you will find various things to help you on your quest. There are potions, enchanted scrolls and weapons. In addition to that some rooms contain pieces of gold. It is the pieces of gold that constitute your score should you not make it out of your quest alive.

Certainly if there are weapons in the rooms for you to pick up there must be an element of danger involved. Where did the weapons come from? Are they left over from other adventures who will never make it back from the Dungeons of Doom? Yes indeed, there are monsters galore in there with you. Everything from bats and emus to orges and ice monsters.

To play Rogue you will need a Color Computer III with at least one disk drive and 128K. On the 128K CoCo III the Dungeons of Doom and all its contents are represented with keyboard characters, such as \$ for some gold and 8 for you. If you happen to have a 512K CoCo III you can use a utility named MAKEGW included on the disk. This will open a high resolution graphics window. Playing the game on this window, gives you graphic depictions for most of the characters in the game. In the graphic window your man on the screen is a man drawn on the screen instead of a number representing you. The monsters remain letters.

As with any game where so much is going on there are a large number of command options. Fortunately they don't all have to be learned at once to start to enjoy the game. Movement can be done by using the arrow keys. This makes starting easy. Picking items up is accomplished simply by passing across the spot in the room where they lay. Of course picking something up doesn't always insure everything is all right. The potions are color-coded. It is up to you to figure out that color-code. Some of the potions will strengthen you, warm you, or put you to sleep. Some can even poison you so be careful. The scrolls are encrypted in a language you will not be able to understand. There is however a scroll of identity which will translate for you. The catch here is that the identity scroll is also encrypted, so you will have to use trial and error to find out which of the scrolls it is. After using a scroll or potion it is gone, but there is usually more than one of any particular item in the maze.

Of course the weapons consist of conventional items, such as swords, armor, maces and assorted other medieval weaponry. In addition to that there are magic items, such as magic wands, rings and staffs. These too, need a bit of experimentation to find out which situation they can best be used in.

If you like a challenge and a game that will keep you occupied for more than a few minutes at a

time then Rogue is the game for you. It is an engrossing game that could keep you tied up for hours. It is available at Radio Shack stores and sells for \$29.95

PUBLIC DOMAIN SOFTWARE

reviewed by Bill Chapple

Public Domain Software is software that is available for everyone at a minimum cost. DungeonMaster Software has put together a disk of programs that should be of interest to color computer users. It is impossible for me to review each program, but I can give you an idea as to what is on the disk. Programs are on both sides and the first side contains:

KINGTUT, SURVIVAL, LLISTER,
CASTLE, CHESS, SILLY, STOCKS,
JOUST, TICTACTO, CIPHER,
STATECAP, FREEWAY, SATAN, GERMS,
OLDHOUSE, REFLEX, SUB, TANKS,
WATOR

The second side contains the following:

FLASEH, COUNT, SKPCOUNT, SPELL,
READLEVL, ATBY, HTO, ANIMALS,
HANGMAN, CAMEL, HAMMURAB,
PUZZLE, MAZE, PLACVLU1,
PLACVLU2, LEMONADE, TCLOCK,
FLASH+, FLASHX, ADDSUBWK,
DAZTERM, MENU, PUZZLE2, GAME,
SORCERER, FILECOPY, READ,
DISKMAP, CAWRITER.

I looked at several of the programs. There are some very good educational programs on the disk that would be helpful for school students. There are also many games and disk utilities. I played a couple of the games and ran a few of the utility programs. They all seemed to work with minimum or no problems. One of the problems with public domain programs is the lack of instructions. However, I did not have any problem figuring out how to run the programs. The cost of the disk is only \$3.00 including shipping and handling.

DungeonMaster Software
P. O. Box 1142
Marshall, NC 28753.

HAM RADIO PROGRAMS

For Radio Shack Color Computers

MORSE - This program allows a key to be pressed and then sounds the Morse equivalent. It also will send random characters. This is an excellent tool for developing code speed for the the Novice, Technician, or General class licenses.

DX - Type in a prefix for a foreign country and have the country displayed.

ANTENNA - An antenna design program that calculates the dimensions for a wide spaced Yagi antenna of up to 4 elements.
Order HR-1 (3 programs) \$11.95

MORSE TERMINAL

When used with an interface this converts your color computer into a Morse Terminal. To transmit just type the Morse characters and the computer keys your transmitter. In the receive mode the computer decodes and displays the Morse characters on the screen. Instructions are included for building an interface with off the shelf parts. HR-2 \$12.95

STATION LOG

Keep a record of your contacts. Just enter the information as it is requested. Items that are the same such as date, frequency, and type of emission need only be entered once and changed as needed. Save and load records to tape or disk. Add to the log and quickly find stations. Print the log to a printer. HR-3 \$9.95

THERMOMETER

Now your computer can give you the temperature in both Fahrenheit and Centigrade. Assembly plugs into a joystick port & consists of a thermistor on a 10' cable for the single unit and a second thermistor on a 20' flat cable for the dual unit. The dual unit can be used to measure inside and outside temperature.
CC-THERM \$12.95, CC-THERM 2 \$19.95.

MEMORY SAVER 2

A battery backup for all color computers. Leave programs in your computer and the Memory Saver will preserve them in case of a power failure. A real time saver for cassette systems. MS-2 \$39.95

WEATHER FACSIMILE (WEFAX)

Draw weather maps on the screen. Feed transceiver's audio into the cassette port. Requires a joystick. WEFAX \$6.95.

HAM RTTY TERMINAL

Uses the cassette port. Requires simple interface to connect cassette audio into the Mic jack and receiver audio into the cassette port. Interface instructions are included. 60, 75, & 100 WPM Baudot.
RTTY \$6.95.

* MORSE KEYER (new) *

Send characters direct from the keyboard or select up to 10 preprogrammed messages to automatically call CQ, CQ DX, First Transmission, Weather, DE your call, etc. Also allows entering the call letters of the station worked and his name which can automatically be sent by pressing only one key. Order the cable below for a super keyer for less than \$25. M-KEYER \$12.95

* KEYER INTERFACE (new) *

Interface cable that connects to the printer port of the color computer & the KEY input of solid state transceivers. Wired for 2 or 3 conductor 1/4 inch plug (state type). Maximum key up voltage is 15 volts. Will not work on vacuum tube transmitters. 6' long, KEY-IN \$12.95

Dynamic Color News on Tape or Disk
\$6.95 each or 6 for \$35 including ship.

AUDIO GENERATOR - Generates exact digital audio frequencies using your computer's crystal as a standard. Audio signal is on the cassette cable. DCN #44.

FREQUENCY COUNTER - Accurately measure audio frequencies up to 12000 hertz. Feed unknown frequencies in on the cassette cable. DCN #45.

TUNING METER- Indicates proper tuning for RTTY and Slow Scan Television. Excellent for use with hardware decoders. DCN #48.

HAM MATH - Solves most problems with circuits, antennas, decibels, etc. An excellent program for studying for ham licenses. DCN #50.

See Dynamic Color News on tape or disk index for additional support programs.

All programs are color computer 3 compatible unless indicated and are on tape or disk. Please specify tape or disk software. A 32K minimum computer is required.

Checks, VISA or MC, Add 93 shipping.

Dynamic Electronics Inc.
P. O. Box 896 (205) 773-2758
Hartselle, AL 35640

HAM RADIO & COMPUTERS

by
bill chapple w4gqc

I have really been enjoying my new antenna system. The day before Thanksgiving I put up a Tri-Band antenna for 20, 15, and 10 meters. This is mounted on top of my house and is only about 20 to 25 feet above the ground. However it really performs well and I have enjoyed operating it. I am sure it would work better if it were a little higher, but I am pleased with its performance.

Last month I presented a program that would calculate power and standing wave ratio (SWR). If the antenna is properly matched to the transmission line then all of the transmitter's power is transferred to the antenna and is radiated. SWR is a measure of the amount of mismatch between the antenna and transmission line. It is actually the quotient of the two impedances. If we let ZA be the antenna's impedance and ZL be the transmission line's impedance then

$$SWR = ZA/ZL \text{ or } ZL/ZA.$$

SWR is always greater than or equal to 1.

On the same mast as my tri-band beam antenna, I mounted a 4 element 2-meter beam for operating on 146 MHz. This is a small antenna compared to my tri-band antenna with a driven element having a length of about 3 feet as compared to 33 feet for the tri-band beam. I have enjoyed talking to people through re-

peaters and also directly. I have a new AZDEN PC-6000 FM transceiver and it is very nice and easy to use. Since antennas for 144, 220, and 440 MHz are small I thought that an antenna program for these frequencies would be of interest. An antenna could be built inside and put up later when the weather is good.

An antenna with wide spaced elements can be built without any critical adjustments being required. The design I have increases the director spacing as more directors are added. In case you are not familiar with a "beam" or "Yagi" antenna, let me give you the basics.

A dipole is used for the driven element. A boom is a rod or pole onto which the elements are parallel mounted. A reflector is mounted parallel to the dipole. Directors are then mounted on the boom on the opposite side of the reflector. The director nearest to the driven element (dipole) is called the first director. The remaining directors are numbered second, third, etc.

The program gives the results in inches and will also print them to a printer. I suggest you refer to an antenna handbook for construction details. Try different numbers of elements and you can see how the size changes.

I have other subjects that I am working on for this series. Next month I will have something

different. I hope you enjoy designing your own VHF antennas.

VHF ANTENNA DESIGN PROGRAM

```

1 CC=PEEK(33021)
2 IF CC=50 THEN GOSUB 1200
5 CLS
10 PRINT~VHF YAGI ANTENNA DESIGN
~
20 PRINT~BY BILL CHAPPLE W4GQC
22 PRINT~cOPYRIGHT (c) 1989
25 PRINT~DYNAMIC ELECTRONICS INC
30 PRINT~THIS DESIGN ALLOWS EACH
40 PRINT~ELEMENT TO BE SHORTENED
BY
50 PRINT~5% OF THE PREVIOUS ELEM
ENT.
60 PRINT~ALSO THE SPACING INCREA
SES
70 PRINT~BY 10% AS ELEMENTS ARE
ADDED.
80 PRINT~DIMENSIONS ARE GIVEN IN
INCHES.
100 INPUT ~PRESS ENTER TO CONTIN
UE~;XX
110 CLS
111 PRINT~ELEMENTS ARE ARRANGED
AS FOLLOWS
112 PRINT~1 REFLECTOR~:PRINT~2 D
RIVEN ELEMENT
113 PRINT~3 FIRST DIRECTOR~:PRIN
T~4 SECOND DIRECTOR
114 PRINT~5 THIRD DIRECTOR~:PRIN
T~6 FOURTH DIRECTOR
115 PRINT~MORE DIRECTORS CAN BE
ADDED.
120 PRINT~PRESS P TO PRINT RESUL
TS
130 PRINT~TO A PRINTER~
140 P$=INKEY$:IF P$="" THEN 140
150 IF P$="P" THEN PRINT ~PRINTE
R ON~ ELSE PRINT ~PRINTER OFF
~
152 IF P$="P" THEN PRINT#-2,~VHF
ANTENNA DESIGN PROGRAM BY BI
LL CHAPPLE W4GQC
160 INPUT~ENTER FREQUENCY IN MHZ
~;FR
165 IF P$="P" THEN PRINT#-2,~FRE
QUENCY="FR
170 INPUT~ENTER NUMBER OF ELEMEN
TS~;EL
175 IF P$="P" THEN PRINT#-2,~NUM
BER OF ELEMENTS="EL
180 WL=11808/FR 'CALCULATE WAVEL
ENGTH
190 DR=5600/FR 'DRIVEN ELEMENTH
IN INCHES
195 X=DR:X$="DRIVEN ELEMENT LENG

```

```

TH~:GOSUB 900
200 RF=DR*1.05 'REFLECTOR LENGTH
205 X=RF:X$="REFLECTOR LENGTH~:G
OSUB 900
210 SP=WL*.2 'REFLECTOR SPACING
212 X=SP:X$="REFLECTOR SPACING~:
GOSUB 900
220 PL(1)=DR*.95 'FIRST DIRECTOR
LENGTH
225 X=PL(1):X$="FIRST DIRECTOR L
ENGTH~:GOSUB 900
230 SP(1)=WL*.2 'FIRST DIRECTOR
SPACING
235 X=SP(1):X$="FIRST DIR SPACIN
G~:GOSUB 900
240 'CALCULATE REMAINING DIRECTOR
S
245 LN=RF+SP(1)+SP(1)
249 '
250 FOR I=2 TO (EL-2)
260 PRINT:IF P$="P" THEN PRINT#-
2,~ "
262 PRINT~DIRECTOR NUMBER~;I
265 IF P$="P" THEN PRINT#-2,~DIR
ECTOR NUMBER~;I
270 SP(I)=SP(I-1) *1.1 'DIRECTOR
SPACING
277 X=SP(I):X$="DIRECTOR SPACING
~:GOSUB 900
280 PL(I)=DR*(100-(4+I))*0.01 'DI
RECTOR LENGTH
287 X=PL(I): X$="DIRECTOR LENGTH
~:GOSUB 900
290 LN=LN+SP(I)
300 NEXT I
310 X$="BOOM LENGTH~:X=LN:GOSUB
900
320 PRINT~PRESS Y TO RERUN THE P
ROGRAM~
330 P$=INKEY$:IF P$="" THEN 330
340 IF P$="Y" THEN RUN
350 END
899 '
900 'PRINT SUBROUTINE
905 X=INT (10*X+5)/10
910 PRINTX$;X
920 IF P$="P" THEN PRINT#-2,X$;X
950 RETURN
1200 PALETTE 12,63:PALETTE 13,0:
PALETTE 8,63:PALETTE 0,0:CLS1
1210 PRINT~1 32 CHARACTER WIDTH
1220 PRINT~2 40 CHARACTER WIDTH
1230 PRINT~3 80 CHARACTER WIDTH
1240 INPUT~ENTER NUMBER FOR WIDT
H~;N
1260 IF N=2 THEN WIDTH 40
1270 IF N=3 THEN WIDTH 80
1275 CLS1
1280 RETURN

```

editor's comments

This issue marks the end of five years of continuous publishing of Dynamic Color News. There have been gradually changes since our first issue. For one thing we started out with a newsletter of only a few pages. Now we have a magazine in the 40 to 50 page range.

Computers have changed since our first issue. For example Radio Shack was selling a 4K color computer for \$399. These could be upgraded to 16K. Another set of 16K chips could be soldered on top of these to give 32K. Most programs would run with this amount of memory. In fact most programs today do not require over 32K except special ones for the color computer 3.

In the earlier days printers and disk drives were expensive. There were numerous programs on tape and a number of people still use tape as their storage medium. I have found the tape software to be reliable and good. However you should never use the cheap tapes as your programs will be lost as the tapes deteriorate with time. I remember loosing a number of programs on tapes before I purchased my first disk drive. I had saved the programs on the \$3/1 tapes. So if you use tapes use high quality audio or computer tapes. The shorter tapes seem to work the best.

There have been tremendous improvements in printers over the past few years. My first printer was an Epson MX-80 and cost about \$450. It developed a problem with one of the print

striker drivers and burned out a print head. I bought a replacement and it too failed. With the cost of print heads near \$100, it was cheaper to buy a new printer than try to repair the Epson. Now for around \$200 a printer can be purchased that outperforms my MX-80 and has near letter quality print.

Disk drives have become very popular for saving programs. I remember when a pair would cost about \$500. Two half height 5 1/2 inch drives can now be purchased for \$300 or less. The trend is towards hard drives. These are a little expensive for the color computer but are coming down in price.

Most computer products have decreased in price over the past few years. Memory chips are an exception and began a constant upward slope until they peaked out last Summer in the \$14 range. They have come down some now but still cost \$10-\$12 depending upon the speed required. This was because of a United States tariff against Japanese semiconductors.

As I look back over last year's accomplishments, I am pleased with the progress that has been made with computers and our production of Dynamic Color News. We still have the same objectives that we had with our first issue and have some very good supporters. Dean and I want to thank each of you for your support and we appreciate any comments or suggestions you give. We want to wish each of you a happy and successful new year.

QUESTIONS & ANSWERS

These are questions that have been asked us. If you have a question about a computer subject, or if you have an answer to a question, we would like to hear from you

Gentlemen:

I have just purchased a 2400 baud modem along with your Dyterm-2M (M-2400) and according to the instructions I will need some type of word processing program to 'clean up' any programs that I may download in ASCII from a bulletin board.

Do you have any such programs available for purchase or can you recommend any that would be useful for this purpose? I am using a CoCo II with tape storage. Thanks

John Fulton

John we will present a program in our February 1989 issue that will allow you to clean up ASCII basic files. Basic programs require numbers at the start of each line. If a line does not have a number at the start then the program will not load and an error will be printed on the screen. Telewriter and VIP word processors will work and of course there are others. In the article we will show how to do this with the program we present and with a word processor. Thanks for your letter.

+ + +

Dean,

Not too long ago I wrote inquiring about converting graphic programs from one form, or another, to usable form for the

Cocomax III programs.

Since that letter was written, I have determined that I DO have a system which will do the job, fairly close to what I want to do.

The route I follow is as follows: First, I have a conversion program called "RENAME". I load and run this program. The procedure calls for then loading any machine language graphics ("BIN") and then follow the menu in the rename program. (Rename the "BIN suffix to the MAX suffix, which in turn will allow the pix to be run on either Cocomax or Cocomax III.) One step which I added was to add lines enabling the conversion from BIN to CM3, which were not in the original program. This however results in a Mexican standoff as the pix which turn out lock up the Coco III. Use the BIN to MAX section!

The next step calls for the loading and running of a program built into the Cocomax III program --Run" Translat". This, in turn will allow the transposition of a M/t program into a useful form for "CM3" which is usable on Cocomax III. A graphic example of a converted pix to a Cocomax III picture is enclosed.

I find that there is room to slightly modify the ensuing program a bit, to possibly correct a minor flaw in it-- for the Cocomax III portion, but it works fine on the Coco II, as is. I have two Coco III machines so I know that the procedure works for me. One added improvement would be the ability to include BASIC graphics in this program, which it will not do as yet. (At least I haven't been able to convert them.)

Since there is a flaw in the

graphic conversion, even though the system works as I have it, maybe someone will experiment further and will have a better idea, or maybe a solution easier to follow or even an improvement on what I have.

The Christmas Season is a wonderful time of the year but it does have its limitations, especially on a limited income! Recently I became very, very much acquainted with with Static Electricity on my two Coco III machines. I tried just about everything I know to clear it up, even having both units sent back to the Service Center for checking over--NOTHING wrong. As a last resort, I found out one thing. In trying to protect my rigs from dust and/or other unwanted pollution, I had used, as a dust cover, parts of an old air conditioner cover (made of a grey cloth-like material and with a backing on it, cut into squares of the appropriate size to cover the machines, printer and RGB. The moment I did two things--get rid of the covers, first, and then spray some anti-static spray on my desk--VOILA, success!

Lou Braun

Lou thanks for the information on graphics conversion and your solution to the static problem. Your letter is very informative and I am sure our readers will appreciate your writing. We all have trouble with Christmas bills.

+ + +

Dear Bill,

I like yoyur magazine very much. I enjoy the Ham section the best. besides using my Coco 3, I like to listen to my shortwave receiver. I use the RTTY program and really get a kick out of it. My question is, can I purchase the Morse Termi-

nal all ready assembled? I do not have any electronic experience. This is something I have been wanting since I first read the article. I have showed the diagrams to the members of my Coco Club but there is no one who can do the assembly for me.

Before using the Coco I had a Timex-!000 computer. There was a M.L. Program that I had that decoded Morse code from the radio. It worked O.K. but not that great. Do you know of any such program for the Coco 3

Thanks

Donald Nelson

Donald thanks for your letter. We have the Morse keyer interface with software which is half of the Morse terminal. We have also had other people inquire about purchasing it wired from us. I am working on a simple interface for receiving Morse code and would like to have a plug in assembly like we have for the Morse keyer. I hope to have this completed soon and then we will come out with a new Morse terminal with an interface that we can easily build.

There are commercial units that you can purchase that convert Morse code to ASCII. These are relatively expensive and there may be a software problem since most software for them are for MSDOS and Commodore computers. A terminal program may be required for use with them. I am going to try to complete our unit soon as we have had many request for this.

+ + +

If you have not written and have a problem I would like to hear from you. Or if you had rather call, I can usually be reached in the evenings and on weekends. - Bill

AUTO - EXEC

Make ML Programs Execute After Loading

by
Doug Canfield

One desire that we have probably all felt, and that a reader addressed in the last issue, was the ability to load in a machine language program, and not have to type "EXEC". I have a few programs that do that, and so after digging into them to see how it was accomplished, I decided to write a program to convert any machine language program to make it auto-executing.

The secret of an auto-execute is to not only load your program's code into its place in memory, but also to load some code into the memory that is normally used by the BASIC operating system in your computer. You can trick the computer into jumping into your program, instead of processing its own normal needs.

There are probably several places into which this "trick" code can be loaded but the most convenient place that I have found occurs at locations 360 and 361 (Hex &H168 and &H169) in lower memory. After the computer faithfully completes your "LOADM" instruction, it looks to this place in memory to find the location of the code it needs to help print the familiar "OK" on your screen. If we cause this location to steer it to the beginning of our program that it has just loaded in, then guess what it does instead?

If you are writing your own machine language programs with an assembler (such as DISK EDTASM), then there is nothing simpler. You only need something like the following:

```
100   ORG   $168
110   FBD   $XXXX
```

WHERE "\$XXXX" is the hexadecimal location at which your program begins execution. You must also include, at the beginning of your program something similar to the following:

```
140   CLR   $FF40
150   LDD   #$CC1C
160   STD   $168
```

Line 140 turns off the disk drive, (which will still be spinning), and you need lines 150 and 160 if you intend to use the ROM routine in memory to print text on the screen. This is what &H168 and &H169 contain initially.

But we're probably not using an assembler, and this won't help us with the programs that we already have anyway. As the assembler would normally write the correct codes on the disk, AUTOEXEC must do all these things instead.

AUTOEXEC actually causes the computer to load code into two additional places in memory. We don't want to jump directly into the program from &H168 since it might use &H168 itself, and would keep jumping to its own beginning if we didn't restore &H168,&H169 to their proper values. We need to jump to another short routine that will fix &H168,&H169, turn off the disk drive, and then jump into the desired program.

If you look at line 470 in the AUTOEXEC program you will see that N\$ represents the exact code that we want to put on the

disk. The first 14 characters are the code to tell the computer to load hexadecimal \$0152 into locations &H168, &H169. &H152 is where we will put our short program to accomplish the little routine outlined in lines 140-160 above. This code is contained in the next 34 characters of N\$. XEQ\$ represents the location in memory where the program that we are auto-executing starts. The last 10 characters are what the computer expects to see at the end of a machine-language program. It contains the execution address of the program.

The code at &H168,&H169 makes the computer jump to the code at &H152-&H15D which makes the computer jump into the final program. AUTOEXEC also patches up the directory so that the computer knows to load in the additional code we are placing on the disk.

There are three possible errors which AUTOEXEC might generate. The first, a "FILE STRUCTURE ERROR" means you are using a garbled up disk. The second and third, "BYTE UNDERFLOW" and "BYTE OVERFLOW", means that all the information that we need to put on the disk cannot be contained on only one sector. This is a remote possibility, so I did not write AUTOEXEC to accommodate it.

Should you get one of these last two errors, there is still hope! You must 1) know the beginning, ending and execution addresses of your program. (See DEC. 1988 DCN, Pg. 16 for a program which finds them) -- and you must 2) have a program which only loads into only ONE place in memory. (Most games seem to.)

To solve your problem, load your program in using the "LOADM" command, and then resave it using:

```
SAVEM~FILENAME~,BEG-30,END,EXEC
```

Where BEG is the normal program beginning address, END is the normal ending, and EXEC is the execution address.

If you have any comments, suggestions, or --errors, I would much appreciate your letting me know.

AUTOEXEC PROGRAM

```

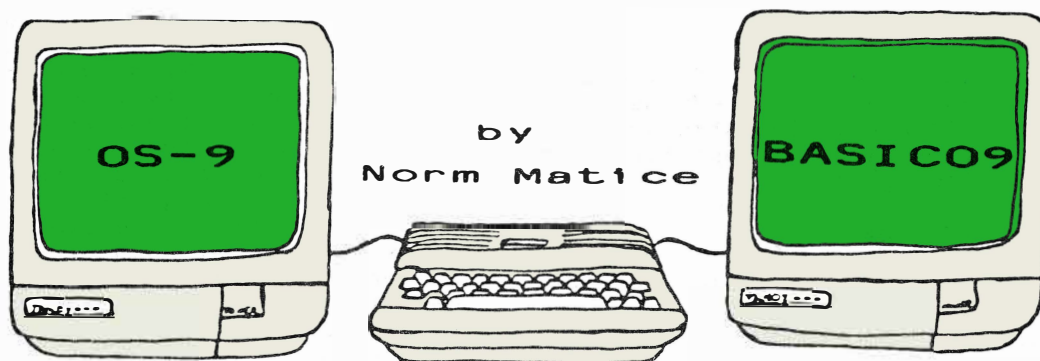
1 ' AUTOEXEC
2 ' BY DOUGLAS CANFIELD
3 '
4 ' ROUTE 1, BOX 86
5 ' FALKVILLE, ALABAMA
6 ' (205) 784-5656 35622
7 '
8 ' COPYRIGHT (C) 1989
9 ' DYNAMIC ELECTRONICS INC.
10 '
15 CLEAR 2000
20 CLS
30 DIR: PRINT@ 148,"FILENAME ?":
   PRINT@ 180,"";
40 LINEINPUT"";F$
45 'MAKE X=THE NUMBER OF CHARS
   IN FILENAME
50 FOR X=1 TO LEN(F$)
60 X$=MID$(F$,X,1)
70 IF X$="." OR X$="/" THEN 100
80 NEXT X
90 XT$="": GOTO120 'ASSIGN DUMMY
   EXT. NONE WAS SPECIFIED
95 'ADD SPACES TO MAKE EXT. 3 CH
   ARS LONG.
100 XT$=RIGHT$(F$,LEN(F$)-X)
105 'ADD SPACES TO MAKE FILENAME
   8 CHARS LONG.
110 XT$=XT$+STRING$((3-LEN(XT$))
   ,32)
120 F$=LEFT$(F$,X-1)
130 F$=F$+STRING$((8-LEN(F$)),32
   )
135 'SEARCH DIRECTORY FOR GIVEN
   FILE
140 FOR SCTR=3 TO 11
150 DSKI$ 0,17,SCTR,A$,B$
160 C$=A$+LEFT$(B$,127)
170 FOR N=0 TO 7
180 PLC=N*32+1
190 NAM$=MID$(C$,PLC,8)
200 EXT$=MID$(C$,PLC+8,3)
210 IF NAM$<>F$ THEN 230
220 IF XT$="" OR XT$=EXT$ THEN 3
   00
230 NEXT N

```

```

240 NEXT SCTR
245 'GIVE MESSAGE THAT FILE WASN
'T FOUND
250 IF XT$="" THEN XT$="****"
260 CLS: PRINT: PRINT " ";F$;
". ";XT$;" NOT FOUND"
270 PRINT: PRINT " PRESS ANY K
EY TO CONTINUE"
280 IF INKEY$="" THEN 280
290 GOTO 20
300 CLS: PRINT@ 197,"USING: ";N
AM$;".";EXT$;"'"
305 ' GRAN (FOR NOW)=THE GRANULE
THAT THE FILE STARTS AT.
310 GRAN=ASC(MID$(C$,PLC+13,1))
315 'BYTES=THE NUMBER OF BYTES
IN THE LAST SECTOR OF THE FIL
E
320 BYTES=256*ASC(MID$(C$,PLC+14
,1))+ASC(MID$(C$,PLC+15,1))
325 'IF OUR ROUTINE WON'T FIT ON
THE LAST SECTOR THEN...
330 IF BYTES+24>256 THEN PRINT"BY
YTE OVERFLOW": END
335 'IF ALL THE INFO. WE NEED IS
NOT ON THE LAST SECTOR THEN.
..
340 IF BYTES<5 THEN PRINT"BYTE U
NDERFLOW": END
345 'GET THE FILE ALLOCATION
TABLE
350 DSKI$ 0,17,2,X$,Y$
355 'FIND THE TRACK AND SECTOR
AT THE END OF THE FILE.
360 LAST=ASC(MID$(X$,GRAN+1,1))
370 IF LAST<69 THEN GRAN=LAST: G
OTO 360
380 IF LAST>&HCO AND LAST<&HCA T
HEN 390 ELSE PRINT" FILE STRU
CTURE ERROR!! SORRY.~": END
390 LAST=LAST-&HCO
400 IF GRAN/2=INT(GRAN/2) THEN T
RACK=GRAN/2:START=0 ELSE TRAC
K=(GRAN-1)/2:START=9
410 LAST=LAST+START
420 IF TRACK>16 THEN TRACK=TRACK
+1
425 'GET THE SECTOR WE JUST
FOUND.
430 DSKI$ 0,TRACK,LAST,Q$,R$
440 S$=Q$+LEFT$(R$,127)
445 'XEQ$=THE EXECUTION ADDRESS
OF THE FILE
450 XEQ$=HEX$(256*ASC(MID$(S$,BY
TES-1,1))+ASC(MID$(S$,BYTES,1
)))
455 'XEQ$ MUST BE 4 CHARS LONG.
460 IF LEN(XEQ$)<4 THEN XEQ$="0"
+XEQ$: GOTO 460
465 'THIS IS OUR ROUTINE, IN
HEXADECIMAL, THAT WE WILL
TACK ON THE END OF THE FILE.
470 N$="0000020168015200000C0152
7FFF40CCCC1CFD01687E"+XEQ$+"F
FO000"+XEQ$
475 'T$=OUR ROUTINE IN A FORMAT
THAT THE DISK NEEDS
480 T$="":FOR X=1 TO LEN(N$) STE
P 2
490 T$=T$+CHR$(VAL("&H~+(MID$(N$
,X,2)))
500 NEXT X
505 'BYTE$=THE NEW NUMBER OF BYT
ES THAT WILL BE ON THE LAST S
ECTOR OF THE FILE (OUR ROUTIN
E ADDS 24). Q$ AND R$=THE IN
FORMATION THAT WAS IN THE LAS
T SECTOR, WITH OUR NEW ROUTIN
E TACKED ON THE END.
506 'WE MUST APPROACH THE CREATI
ON OF Q$ AND R$ DIFFERENTLY I
F WE NEED ALL 256 BYTES THAT
CAN FIT ON A SECTOR.
510 IF BYTES+24<256 THEN BYTE$=C
HR$(0)+CHR$(BYTES+24): S$=LEF
T$(S$,BYTES-5)+T$: Q$=LEFT$(S
$,128): IF LEN(S$)<129 THEN 5
40 ELSER$=MID$(S$,129,LEN(S$)
-128): GOTO 540
520 BYTE$=CHR$(1)+CHR$(0)
530 Q$=LEFT$(S$,128): R$=MID$(S$
,129,BYTES-133)+T$
535 'C$=THE THE DIRECTORY SECTOR
WE ARE USING.
540 C$=LEFT$(C$,PLC+13)+BYTE$+RI
GHT$(C$,255-(PLC+15))
550 A$=LEFT$(C$,128): B$=RIGHT$(
C$,127)
560 PRINT: PRINT " REWRITE: ";N
AM$;".";EXT$;"'? (Y/N)~": PRIN
T
570 X$=INKEY$: IF X$="" THEN 570
580 IF X$<>"Y" THEN PRINT TAB(9)
"***ABANDONED***": GOTO 620
585 'THESE TWO LINES PUT OUR
FINISHED PRODUCT ON THE DISK
590 DSKO$ 0,TRACK,LAST,Q$,R$
600 DSKO$ 0,17,SCTR,A$,B$
610 PRINT" -=FILE REWRITTE
N=-
620 PRINT: PRINT TAB(7)"DO ANOTH
ER? (Y/N)"
630 X$=INKEY$: IF X$="" THEN 630
640 IF X$<>"Y" THEN END ELSE 20

```



MORE MULTITASKING

As I promised last month, we will look at ways to set the priority of a process. If for example you wished to speed up your computer and slow down your terminal we can change the amount of cpu time that both of them get. This will work with other processes as well. If you have a program in one window that needs more CPU time than something happening in another window, or if you have a task running in the background you may want to alter the CPU times for those processes.

Editors Comment: The microprocessor is the central processing unit (CPU). It can only do one task at a time so it has to share its time between the computer and terminal. Also when you are doing multitasking the time has to be shared among the tasks.

The first thing to do is to find out how much time each process is allocated and what its process id number is. The command to do this is the PROCS command. Type PROCS at the OS-9 prompt and it will return the following information on the currently running processes, Id, PId, User Number, Pty, Age, Signal, Mem Size, Stack Ptr, Primary Module. If you have just entered the command you should have gotten back reports on at least the SHELL you are using and the PROCS command itself.

The items we are most interested in are Pty and Id. It is these two parameters we will need to change to set the CPU time allotted to a process. As you can see on your screen OS-9 gives each process a Pty (priority) of 128. The range goes from 1 (being the lowest priority) to 255 (being the highest priority). Why the numbers 1 to 255? Those are the numerical values that can be represented with an eight bit byte (and 0). So if you want one process to have the highest priority available you would assign it a value of 255. The other thing we need to know (besides the range) to set this priority is the process Id number. This is what is in the first column of the PROCS output. If you look in the column labeled User Number you will see a column of zeros. This represents the superuser, which is you unless your logged on a terminal as a different user.

If you are the superuser of a system and you want to run the PROCS command on

every process being used at that time you would use the command PROCS E. Now that you have a list of everybody's processes, their Ids and their priorities, we can now set new priorities. To do this we will use the SETPR command. The syntax is SETPR Id Pty. As an example if we wanted to change the amount of CPU time our main shell got in a multitasking environment, such as with a terminal connected, we would enter SETPR 2 255 (on my computer 2 is the Id for the shell). This would insure that my shell had the highest priority and thereby speed up the processing on my screen.

We could also use SETPR to lessen the priority of the shell running in the terminal. SETPR 7 1, would give the terminal shell the lowest priority, therefore less CPU time is required. Doing this helps to offset the faster terminal we encountered last month when hooking a terminal on level II. Just remember to use PROCS to get your Id numbers because they may not coincide with the ones in my example.

MERGE COMMAND

Lets continue on with a look at the MERGE command. MERGE does as its name implies, it merges separate files into one bigger file. To see this in action lets build a couple of smaller files and then merge them together. In the first file we will put half of a Ben Franklin quotation and the other half we will put in the second file. We will then MERGE the files and see if we get a whole quotation. Build the following two files:

```
FILE1
? EARLY TO BED
? AND EARLY TO RISE
?
```

```
FILE2
? MAKES A MAN
? HEALTHY, WEALTHY, AND WISE.
?
```

Now to get these two files into a combined file we simply need to redirect the output of the MERGE command to the new file. The following command will do the job.

MERGE FILE1 FILE2 >QUOTE

If you list the file QUOTE you will see that the two files do indeed make up the new file. Also note that the order in which we do our merging is important. If we try the following command

MERGE FILE2 FILE1 >REV

and then list REV, we will see that the files are inserted in the reverse order of what we wanted.

The use of the redirection symbol in this commands means that we are not limited to just merging files into bigger files. We can also merge files to different locations, such as the printer. Instead of >REV above we could have used >P to send the output to a printer, or >/T1 for a terminal we might have hooked up.

One other thing to note is that even though we merged our two small files into one larger file, we still have our two original files in the directory. The two small files were copied into the larger file and were therefore not destroyed in the process. If after merging files the original files can be delete them to free up disk space, or they can be retained as backup files. Its sort of like having a backup copy. You are free to select the way that suits your needs.

BASIC09**COMPARING BASIC 09 & ECB**

For some time now we have been looking at the differences between Extended Color BASIC and BASIC09. We have seen some of the new programing commands that BASIC09 offers us over Extended BASIC. I think we are now ready to test drive BASIC09 and check out the added horsepower of this version of BASIC.

One of the main complaints about higher level languages such as BASIC is that they are slow, when compared to an assembly language program. This is because most versions of BASIC (including Extended BASIC) are interpreted languages. What this means is that each time the program is run, the BASIC language is translated into a set of assembly language commands for the computer, much the same as an interpreter does with a foreign language.

It is obvious that a conversation goes much quicker if two people are speaking the same language, than if a third party has to repeat everything that is said. The same is true of your computer. Unfortunately the computer's language is a little harder than most. The other scheme for high level language translation for your computer is the use of a compiler.

A compiler translate your code into assembly language commands once and leaves them in that form. Now instead of translating your program each time you run it, it's already in a language the computer will be able to understand and act on quickly. Compiling is the scheme that

BASIC09 uses on your procedures. As you enter the procedure it is compiled.

Lets try two similar programs and check out the speed difference between the two types of program translation. The first listing below is in Extended Color BASIC. Type it in and run it before you boot up OS-9. Our main interest in the program is lines 90 and 100. These two lines have a simple loop that counts from 1 to 10000. This program takes about 22 seconds from the time the message is started until the word BASIC appears to complete it.

Listing 2 is a similar program in BASIC09. I used the WHILE/DO/ENDWHILE in the first loop just for variety's sake. The section we use as a judge of execution speed is the same. When we run the BASIC09 procedure, the time between when the message first appears and when it finishes is less than one and one half seconds. Quite a nice improvement over Extended BASIC.

Now all those application programs, arcade games or adventures that you'd have written in BASIC, but thought were too slow, have a second chance. That doesn't mean BASIC09 is perfect though. It is still slower than assembly language and even with its extra features it still lacks some of the niceties of Extended BASIC. If you look at the listings you'll notice that the placement of text on the screen would have been easier with the use of the PRINT@ command. BASIC09 doesn't have that feature, unless you are in a graphic window.

Still it is certainly a more powerful, faster language than Extended BASIC. Besides as you get used to using it you won't notice the few things it doesn't have and come to appreciate the many things it does.

LISTING 1:

```
10 CLS
20 X=0
30 FOR X=1 TO 6
40 PRINT
50 NEXT X
60 PRINT TAB(15);"HOW"
70 PRINT TAB(13);"SPEEDY"
80 PRINT TAB(15);"IS"
90 FOR Y=1 TO 10000
100 NEXT Y
110 PRINT TAB(14);"BASIC"
120 END
```

LISTING 2:

```
PROCEDURE SPEEDTEST
SHELL "DISPLAY C"
DIM X,Y:INTEGER
X=0
WHILE X<6 DO
PRINT
X=X+1
ENDWHILE
PRINT TAB(15); "HOW"
PRINT TAB(13); "SPEEDY"
PRINT TAB(15); "IS"
FOR Y=1 TO 10000.
NEXT Y
PRINT TAB(13); "BASIC09"
END
```

Editor's Comments.
Questions and Answers, and Operating Hints are included in all issues. Product reviews are in issues since #17. Numbers in Parentheses indicate the part of a series. A N indicates the issue is not available on disk or tape. All issues except #1,2,3, 16, & 17 are available on disk or tape.

#1 Feb 84 N
Basic Programming
Programming Theory
Multiprogram Manager
MPM Program Listing
Machine Lang. Prog.

#2 Mar 84 N
Basic Programming
Memory Expansion
Machine Lang. Prog.

#3 Apr 84 N
ASCII Part 1
Basic Programming
Machine Lang. Prog.

#4 May 84
ASCII Part 2
ML Programming-
Interrupts
Installing an
Interrupt switch
Utility Program

#5 June/July 84
Powerful Remarks
(Remarks for data)
ASCII Part 3
Uninterrupted Power
Machine Lang. Prog.

#6 Aug 84
Powerful Remarks (2)
Data in Remarks
Check Book Program
Word Processor
Machine Lang. Prog.

#7 Sept 84
Machine Lang. Prog.
Writing Position
Independent Code
Powerful Remarks (3)

#8 Oct 84
Basic Programming
(Data Handling Tech)
Machine Lang. Prog.
(Memory Searching)

#9 Nov 84
Computer Sound (1)
Basic Programming
(Sorting)
Ball Team Sort Pgm

#10 Dec 84
Random Numbers
Computer Sound (2)
Sound Learning Pgm
Sound Demo Program

#11 Jan 85
Random Numbers (2)
Card Shuffling
Computer Sounds (3)
Machine Language Sub
Sound Program Disc.

#12 Feb 85
Large Memory Pgas (1)
Computer Graphics (1)
Video Reverser (hard)

#13 Mar 85
Large Memory Pgas (2)
Computer Graphics (2)
Writing Programs (1)

#14 Apr 85
Large Memory Pgas (3)
Computer Graphics (3)
Graphics Demo Program
Writing Programs (2)
Print Demo Pgm

#15 May 85
Writing Programs (3)
Gas Mileage Program
Large Memory Pgas (4)
Data Move Program
Computer Graphics (4)
Graphics Demo Program

#16 June 85 N
Writing Programs (4)
Grade Book Program
Computer Graphics (5)
Character Gen. Pgm
Large Memory Pgas (5)
Address File Program

#17 July 85 N
Large Memory Pgas (6)
64K RAM Program
Writing Programs (5)
Alarm Clock Program
Computer Graphics (6)
Character Gen. Pgm.
CoCo Heat Problem

#18 Aug 85
Writing Programs (6)
Address File Program
Large Memory Pgas (7)
Study Program
Computer Graphics (7)
Line Demo Program

#19 Sept 85
Writing Programs (7)
Fast Food Program
Computer Graphics (8)
Bar Graph Program
Vector Correction Pgm
Large Memory Pgas (8)

#20 Oct 85
Writing Programs (8)
Word Processor Pgm
Bar Graph with
Character Generator
Second Port
Page -1 for 32K + MEM
Large Memory Pgas (9)

#21 Nov/Dec 85
Writing Programs (9)
Check Book Program
Computer Graphics(10)
Circle Demo PGM
Recipe Program
RAM Disk Program
Electric Cost Program

#22 Jan 86
Writing Programs (10)
Inventory Program
Computer Graphics(11)
ARC & Circle Demo PGM
Large Memory Pgas(11)
Ship War (Game)

#23 Feb 86
Writing Programs (11)
File DEMO Program
Basic Basic (1)
Computer Graphics(12)
Draw Demo Program
Interfacing Comp. (1)
Bouncing Ball (Game)

#24 Mar 86
Interfacing Comp. (2)
Electronic Billboard
Writing Programs (12)
Basic Basic (2)
Computer Graphics(13)
Draw Program
(Scalling)
Large Memory Pgas(12)
Rasdisk Subroutines

#25 Apr 86
Interfacing Comp. (3)
Writing Programs (13)
Basic Basic (3)
Tanks (Game)
Large Memory Pgas(13)
Upper Memory Program
Computer Graphics(14)
Graphics Programming
(GET & PUT)

#26 May 86
Writing Programs (14)
ML Programming (1)
Interfacing Comp. (3)
Roulette (Game)
Page -1 Pgm Dev.

Basic Program Restore
Large Memory Pgas (F)
Computer Graphics(15)
Graphics Draw Program

#27 June 86
ML Programming (2)
ML Program (Addition)
Page -1
Mem. Peek & Poke Pgm
Writing Programs (15)
Inventory Program
Interfacing Comp. (5)
Chords (Music Pgm)
Computer Graphics(16)
Graphics Draw Program

#28 July 86
ML Programming (3)
ML Add with Carry
Graphics Zoom Program
Writing Programs (16)
File Program
Interfacing Comp. (6)
ASCII Demo Program
Astro Dodge Game
Computer Graphics (F)

#29 Aug 86
ML Programming (4)
ML Subtract Program
Interfacing Comp. (7)
Organize VCR Tapes
Ham Radio & Comp. (1)
Morse Code Program
Disk Disassembler
Basic Prog. (17)

#30 Sept 86
ML Programming (5)
ML Data Move Program
Disk File Utility
Basic Programming
File Program
Ham Radio & Comp. (2)
Antenna Design Pgm
Interfacing Comp. (8)
ML Output Subroutines

#31 Oct 86
Money Chase (Game)
ML Programming (6)
Multiple Choice Test
Basic Programming
Address File Pgm (1)
Introduction to OS-9
Interfacing Comp. (9)
Ham Radio & Comp. (3)

#32 Nov 86
Star Constellations
ML Programming (7)
CoCo 3 (1)
Basic Programming
Address File (2)
Duellling Cannons
Ham Radio & Comp. (4)
DX Program (Ham)
Interfacing Comp.(10)
Hardware Interface

#33 Dec 86
ML Programming (8)
CoCo 3 (2)
OS-9
Lucky Money (Game)
Interfacing Comp (11)
DYTERM Terminal Pgm
Basic Programming
(Sorting)
Ham Radio & Comp. (5)

#34 Jan 87
Cassette Control Sw.
ML Programming (9)
Jungle Adventure
Interfacing Comp.(12)
Ham Radio & Comp. (6)
Morse Code Keyer Pgm
CoCo 3 (3)
Basic Programming
Address File & Sort

#35 Feb 87
ML Programming (10)
Interfacing Computers
(Joystick Voltmeter)
Ham Radio & Comp. (7)
Scrolling Around
Basic Programming
Address File

#36 Mar 87
OWARE (Game)
Joystick Ohmmeter
EPROMS (1)
ML Programming (11)
CoCo 3 (Lowercase
Char & Rev.Video)
Ham Radio & Comp. (8)
Basic Programming
Invoice Program

#37 Apr 87
Diver (Game)
EPROMS (2)
ML Programming (12)
Using Joystick Port
for Measuring Temp.
Ham Radio & Comp. (9)
CoCo 3 (Error Trap)
Basic Programming

#38 May 87
Joystick Digital
Thermometer
Accounts Payable
(Business Pgm)
EPROMS (3)
ML Programming (13)
CoCo 3 Memory Manager
Ham Radio & Comp (10)
Basic Programming
(Fast Sorting)

#39 June/July 87
Genealogy Program
ML Programming (14)
Measuring Light with
Joystick Ports
EPROMS (4)
CoCo 3 Graphics Demo
Ham Radio & Comp.(11)
Horse Terminal Pgm.
Calendar Program

#40 Aug 87
Job Costing Program
ML Programming (15)
Basic Programming
Ham Radio & Comp.(12)
Ham Radio Log
Compound Interest Pgm
CC-3 Hi-Res Graphics
Save Program
Dog Race Program

#41 Sep 87
Astro Dodge Game
ML Programming (16)
Reformatting Data (1)
Meteors Program
Computer Terminology
Ham Radio & Comp.(13)
Relay Interface
(Hardware Project)

#42 Oct 87
Taking Control (1)
(Basic Programming)
ML Programming (17)
Disk Cataloger Pgm
Reformatting Data (2)
Parachute (Game)
Ham Radio & comp.(14)
HAM RTTY Program
Print Program

#43 Nov 87
Save the Maiden (G)
Taking Control (2)
ML Programming (18)
Reformatting Data (2)
Music Program
HAM Radio & Comp.(15)
(Packet Radio)

#44 Dec 87
Kingspede (Game)
Taking Control (3)
Printer Utilities
Ham Radio & Comp.(19)
Audio Generator Pgm
Reformatting Data (3)
Screen Dump Program

#45 Jan 88
Living Maze (Game)
ML Programming (20)
Ham Radio & Comp. (17)
Frequency Counter Pgm.
Taking Control (4)
Reformatting Data (6)
FANTASY2 Music Pgm
Pmode 4-HRES

#46 Feb 88
Using ROM Routines
Taking Control (5)
OS-9 (New Series)
BARRACAT (Card Game)
Ham Radio & Comp (18)
Improved Ham Log Pgm

#47 Mar 88
Using ROM Rout. (2)
Taking Control (6)
OS-9 & Basic 09
Coast-Coast (Game)
Logic (Ed. Game)
Forth Prog. Lang.
ML Loader Pgm.
Ham Radio & Comp.(18)
WEFAX -Weather Pgm

#48 Apr 88
Marriage of ML & Bas
Area Code (Program)
OS-9 & Basic 09
Taking Control (7)
(Basic Programming)
Forth Programming (2)
Fast Dir (Disk Pgm)
Ham Radio & Computers
Tuning Meter Program
Atlanta (Picture)

#49 May 88
Advance (Game)
Marriage of ML & Bas
Superspell (Ed. Pgm.)
OS-9 & Basic 09
Sink the Ships (Game)
Basketball Math (G)
Taking Control (8)
Ham Radio & Computers
Ham Math Program

#50 June 88
OS-9 & Basic 09
Media Master (Pgm)
Mysterious Island (G)
Taking Control (9)
Basic Prog.
Improved Word Proc.
Forth & Forth-Editor
Ham Radio & Computers
Improved RTTY Program
Airplane (Game)

#51 July 88
Using Interrupts
Maze (ML Game)
Forth Background
Taking Control (10)
Player Guitar (Prog)
Kwik Kopyur (Disk Ut)
Ham Radio & Computers
Audio Squelch Prog
Hardware
Motor Jump (game)
Bustout (game)

#52 Aug 88
Trio (Three Prog)
COCO III (new series)
HI-Res Screen Saving
With Prog
EDT-MAC (AL Ut.)
Taking Control (11)
Teacher Grade Prog
Ham Radio & Computers
Handy Ham Prog
Hardware (RS-232 Switch)
U.S. States (Prog)
Loan Program

#53 Sept 88
OS-9 & Basic 09
CoCo III (part 2)
Educational TRIO
Neighboring States
Basic Programming
(Direct Access Files)
Direct Access File Pgm
Alien (CoCo 3 game)
Ham Radio & Computers
Bible Quiz

#54 Oct 88
OS-9 & Basic 09
COCO III (Part 3)
(High Resolution Graph)
Squares (game)
Basic Programming
(Using Disk Files)
Grade Program
Flashcard (ED. Prog)
Ping Pong (game)
Ham Radio & Computers
Improved Morse Keyer Pgm
World Map (pgm)

#55 Nov 88
OS-9 & Basic 09
COCO III (part 4)
(High Resolution Text)
Super Blitz (game)
Disk Drive Problems
(with double sided ut)
Ham Radio & Computers
Horse Keyer Interface)
Forward (game)
Basic Programming
Disk I/O Pgm
Drill (Math Pgm)

#56 Dec 88
OS-9 & Basic 09
(Using Multiuser Feature)
Screen Dump Pgm
COCO III (part 5)
Assembly Language (new)
Basic Programming
(disk I/O)
ML Address Finder Pgm
Ham Radio & Computers
SWR & Power Pgm
Hangman (Word Game)
TIC-TAC (game)

**BULLETIN BOARDS
COMPUTER CLUBS**

**COLOR COMPUTER
CLUBS**

If you want a free listing send us the information. These listings will be kept current. Please let us know of any errors or discontinued bulletin boards or clubs.

BULLETIN BOARDS

Come and enter the realm of the Dungeon BBS of Newport NC. Supporting 300/1200/2400 Baud, Online 24 Hours. The board features Hi-Res graphics, forums, special Interest Groups, Multi Message Bases, News and Information, Market and Advertising and Upload and Download areas for CoCo IBM, Atari, Apple, Tandy, and Macintosh. 30 MEG Capability. Call 919-726-9737.

Parms: 8-N-1. Chuck Katsekas
410 Scott Dr
Newport, NC 28570

The COCONET BBS Central Oklahoma Computer Org. in Oklahoma City, OK area. Operates 24 hrs per day with no password or connect fees. Contains almost 2 megabytes of COCO software and related files. (405) 376-1494

The Computer Cellar, 300/1200/2400 Baud 24 hrs. /day (319) 277-0646 Kevin Langenwaller
2605 Eddie St
Cedar Falls, IA 50613

The CoCo Nut Tree, 300/1200 Baud, 24 hrs, 7 days a week, no parity, 7 data bits, 1 stop bits, echo off. (216) 530-6809

The "Manton Modem" BBS 300 Baud, full Duplex 7E1 24 hrs daily. Running on a CoCo 3, 512K OS9 Level 11 with S.D. Roberson's PBBS V.5.0 storage on a 30 meg harddrive. (616) 824-6026

SYSOP- Carl Johnson
6030 N. 43rd
Manton, MI 49663

COCORAMA (312) 307-1519 7 days, 24 hrs 300-1200 Baud

MINDMASTERS DOMAIN (312) 463-8932 7 days, 24 hrs 300-1200 Baud

FAT NEWSLETTER 300/1200 Baud, FAT 1010
Concord Av Piqua, OH 45356 (513) 778-9624

Hot CoCo Users Group, La Porte City, IA 50651
Monthly News Letter & BBS (319) 277-0646

Electronic Info. Network George Matyaszek
Syslink (312) 622-4442 Chicago Syslink Network
1718 N. Long Av Chicago, IL 60639-4321

Benchboard BBS Online Since 1984! 300/1200/2400 Baud, 8-N-1 24 Hrs/Day, 40 Megs online!
Charter member of the Modem Support Group.
(517) 394-2447

Jan Colucci Editor Crier The Color Computer Club, PO Box 478, Canfield, OH 44406

Jack Eizenga, Treasurer & Disk Librarian
Color America 3811 N. Foster Av, Baldwin Park, CA 91706

Mid Iowa CoCo, Terry Simons, Treas. 1328 48th, Des Moines, IA 50311 (515) 279-2576

T-BUG Newsletter, Linda Hapner, 3329 B. Beacon #50, N. Chicago, IL 60064

Glenside Color Com. Club, Serving Chicago & the Western Suburbs, Ed Hathaway (312) 462-0694

Cook Cty Color Computer Club, Serving Chicago and the Southern Sub. Tony Nowakowski (312) 895-0393

Illinois C.C. Club of Elgin, Serving the Northwest Suburbs of Chicago, Tony Podraza. (312) 428-3576

Greater Lansing Color Comp Users Group Dale Knepper, Pres., PO Box 14114, Lansing, MI 48901. Meets every 4th Tuesday at 7:00pm

BACK ISSUES

Back issues of Dynamic Color News are available for \$2.25 each, 3 for \$5 or 12 for \$18 pp.

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

**ADVERTISER'S
INDEX**

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

T & D Subscription Software .	6
SPORTSware	7
Bill Bernico Software	8
RainyDay Software	11
John Galus	14
Dynamic Electronics Inc .	2, 10
15, 16, 18, 20, 21, 22, 23, 30	



DYNAMIC COLOR NEWS

ON Disk or Tape

Now you can have all of our editorials and programs for your tape or disk library. Programs are ready to load and run. We have a variety of programs such as games, geneology, home management, business, and utility programs. The editorials are saved with a 32 column width as a word processor file so you can review them on your screen or print them on your printer. Software is included for viewing the editorials. Combine each part of a series to form a booklet on each subject.

If you are interested in programming then study the examples given in our programming series. Example programs are included on disk or tape. We are covering both basic and assembly (machine language) programming. Suppose you want to use the extra memory in a 64K color computer. Then review the editorials and examples on managing the extra memory and run the memory manager programs.

Do you want to learn to interface your computer using the joystick port? We had a series on this with example programs for making a voltmeter, thermometer, ohmmeter, and light meter.

If your interest is ham radio then we have articles each month since August 1986. We covered Morse code, Antenna design, DX stations, Morse Keyer, Morse Terminal, and Radio Teletype with support programs.

We support the color computer 3 and have given programs for using the memory manager, graphics and error trapping.

All programs are ready to run and complement the editorials in the magazine. We have covered many subjects and there is much more to come. All of our back issues are available on disk or tape. See our cumulative index for a list of subjects. We also have program collections of key programs from past issues. See our advertisement in this issue.

COST

	USA & Can.	Foreign (Air)
1 year	\$60.00	\$75.00
6 months	35.00	49.00
1 month	6.95	8.95

Back issues are at the same rate. See our Cumulative index for subjects.

* Please sign me up for one year for DYNAMIC COLOR NEWS. I want
* to receive instruction on programming, Computer Theory, Operating
* Techniques, Computer Expansion, plus information on New Products,
* Product Reviews, Programs and Answers to Questions.
*
* Magazine Subscription \$18 USA, \$20 Canada \$50 other foreign.
* Back issues \$1.95 each, 3 for \$5, 12 for \$18 including shipping.
* Disk or Tape Subscriptions are \$60 USA & Canada, (\$75) foreign.
* Single Copy \$6.95 (8.95), 6 for \$35 (49.00), 12 for \$60 (75.00)
* Send back issues _____
*
* 10% club discounts for 5 or more subscriptions.
*
* Name _____ Mail payment to
* Address _____ Dynamic Electronics Inc
* City _____ P. O. Box 896
* State & Zip _____ Hartselle, AL 35640
* Enclosed is a check _____ Number _____ Exp. _____
* charge to VISA _____ MC _____

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.

PREMIUM QUALITY DISKS. You don't have to pay a lot for QUALITY disks. Our disks are complete with labels, sleeves, and write protect tabs and work on COCO's and MSDOS computers. These are double sided double density disks and will be replaced if defective. DSK-2 \$4.95 for package of 10. Add \$1.00 S/H. Dynamic Electronics Box 896, Hartselle, AL 35640. (205) 773-2758

RENEWAL TIME?

If 1/89 is beside your name on your address label then your subscription has expired.

Dynamic Color News is now available on tape or disk for \$6.95 for 1 month, \$35 for 6 months, \$60 for 12 months.

DISPLAY ADS

(Rate sheet 3 - March 1988)
Closing 1st of preceeding month.
Example: Ad for March issue should be received by Feb. 1.

Pages	1X	3X	6X	12X
1/4	20	18	16	15
1/2	30	27	25	23
1	40	37	35	33
2	70	65	60	55

We can do titles for your ad in Red, Blue, Green, or Brown. No all one color ads will be accepted. For color ads send artwork for each color and add 30% per color. Example: One page black and red for 3 times costs \$37 + 11.20 = \$48.20 each month for the three monthss.

Artwork must be camera ready and can be enlarged or reduced at no extra cost. Rates are per page or fraction thereof. We can set up your ad for a reasonable price. Enclose payment with ad copy. Contracts are available. Call or write for a contract form. No X-Rated ads.

Dynamic Electronics Inc.
P. O. Box 896 (205) 773-2758
Hartselle, AL 35640

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