NEWS

COLORCUE QUARTERLY

The COMPUCOLOR II Users Group Newsletter

Includes Volumes I.1 through II.1

Price $4.00

Compucolor® Corporation P.O. Box 569, Norcross, Georgia 30091, Tel. 404/449-5879, Twx. 810/766-1581

Vol.I.1 Oct.'78

COLORCUE

$1.00

HELLO   BUENOS DIAS   GODDAG   BONJOUR   GUTEN TAG   A$="HELLO":PRINT A$

READERS--

We are pleased to present the first issue of ColorCue, a newsletter for and about you and your personal computer, the COMPUCOLOR II. A newsletter such as this can serve many purposes, and as it begins to take form over the next few months, your contributions and criticisms will determine its character.

This publication provides a means of communication between us and COMPUCOLOR II owners, and, as things progress, between the owners themselves. We plan to offer explanations of various programming techniques that will help the new user become more adept. In addition, we will discuss features and technical aspects of the COMPUCOLOR II which will stimulate and inform even the most sophisticated owners/users.

In order to serve your needs, it is necessary that you make those needs known to us. We welcome any requests you may have for articles. If you're curious or perplexed about a certain aspect of programming or machine use, chances are that another user is too, and sufficient reader interest will generate a feature in ColorCue. Your suggestions and encouragements are also welcome and we plan to share some of them with other readers.

This paper will keep you abreast of innovations at Compucolor Corporation. We will be reporting on new options, peripherals, and Library Diskettes as they are made available.

This Issue's MENU:

Many of you are eagerly awaiting the next Library Albums. Several new ones, including BlackJack and Personal Finance, will soon be available. If you have any suggestions for programs you would like to see, please let us know. Perhaps a favorite game or financial function would be a useful addition to the COMPUCOLOR II Program Library.

Many programmers, beginners and seasoned pros alike, have questions about the features of the COMPUCOLOR II, such as its use as a terminal, its particular form of BASIC, or its hardware specifications. Your questions have answers in the COMPUCOLOR II Programming and Reference Manual. It is a ready source of information for all who use the machine. It can be purchased through your dealer or ordered directly from us at Compucolor Corporation.

ADVANCED APPLICATION

How to Use The Scrolling Patch

Selective scrolling on the screen of the COMPUCOLOR II can be done quite easily by including a short BASIC routine in your own programs. This routine allows you to scroll a certain portion of the screen independently of the rest of the screen. It is used in several of the COMPUCOLOR programs in the Diskette Library, including STAR TREK. Because the routine is complicated, space prohibits our describing the details of how it works. Instead, we will show you how to use it.

First, initialize BASIC and key in the lines 0, 1, and 64000 through 65410 as listed below. If you plan to use the scrolling routine in a number of your programs, save the partially completed routine on disk so that you do not have to re-enter it each time that you want to use it. To save it, type SAVE"SCROLL" which saves the routine in a file called SCROLL.BAS on your diskette.

```
0 GOTO 65000
1 REM THERE MUST BE A LINE NUMBERED 1:STRING SPACE=25 BYTES
64000 GOSUB 65410
64010 Z=28672+128*Y+X+X:AD=TM+2: GOSUB 65400
64015 Z=128-W-W:AD=TM+25:GOSUB 65400
64020 POKE TM+5,H-1:POKE TM+7,W*(C+1):POKE TM+19,35*(1-C)
64030 RETURN

65000 GOSUB 65410:RESTORE 65030
65030 DATA 33,-1,-1,6,-1,14,-1,17,128,0,25,126
65040 DATA 17,128,255,25,119,35,-1,13,194,-1,-1
65050 DATA 17,-1,-1,25,5,194,-1,-1,201
65055 IF TM>65503 THEN TM=TM-32:GOTO 65080
65060 FOR I=1 TO 32:READ A
65065 IF A>=0 AND A<> PEEK (TM+1) THEN I = 32:TM=TM-32
65070 NEXT
65080 RESTORE 65030
65090 FOR I= 1 TO 32:READ A: POKE TM+I,A-(A<0):NEXT
65100 Z=TM+1: AD=33283:GOSUB 65400
65110 Z=TM:AD=ER:GOSUB 65400
65120 Z=TM+6:AD=TM+30:GOSUB 65400
65130 Z=TM+8:AD=TM+22:GOSUB 65400
65190 CLEAR 25: GOTO 1
65400 ZZ=INT (Z/256):POKE AD,Z-256*ZZ:POKE AD+1,ZZ:RETURN
65410 ER=32940:TM=256*PEEK(ER+1)+PEEK(ER):RETURN
```

To use the scrolling routine, you must specify a number of parameters that define the size and position of the area on the screen that will be scrolled. The 4 determinants are:

X - The x cursor position (0-63) of the upper left hand corner of the area to scroll

Y - The y cursor position (0-31) of the upper left hand corner of the area to scroll

H - The number of lines (>1) in the area to scroll

W - The number of characters (1-64) in each line in the scrolling area

A fifth parameter is necessary in order to specify whether or not the color status information is to scroll. It is defined as follows:

C - If C=0, then the color information does not scroll. If C=1, the color information is scrolled along with the character.

For example, if an 8 line by 16 character area is to be scrolled in the upper right hand corner of the screen, the parameters should be set as follows:

```
X=48: Y=0: H=8: W=16: C=1
```

By executing a GOSUB 64000, these parameters will be set into the assembly language scrolling routine. Thus, whenever a CALL(0) function is executed, the area will scroll up one line. The bottom line of the scrolled area is not spontaneously erased and must be erased by a PRINT statement. A completed scrolling routine should be similar in form and appearance to the one shown below. Lines 10 through 140 are a sample routine.

```
0  GOTO 65000
1  CLEAR 200: REM 200 BYTES OF STRINGS
10 REM SET SCROLLING AREA TO MIDDLE OF SCREEN
20 X=16: Y=8: H=16: W=32: C=1: GOSUB 64000
80 PLOT 27,24,6,32,12: REM SET PAGE MODE, COLOR, ERASE PAGE
90 PLOT 6,6
100 FOR LN= 1 TO 50
110 GOSUB 1000:PRINT "THIS IS LINE" LN
120 NEXT LN
130 END

1000 X=CALL(0)
1010 PLOT 3, X, Y+H-1: PRINT SPC(W)""
1020 PLOT 3,X,Y+H-1
1030 RETURN
```

The area will continue to scroll one line at a time each time line 1000 is executed. Several portions of the screen cannot be scrolled simultaneously, but they can be scrolled one after the other by executing GOSUB 64000 with the appropriate parameters before issuing each CALL(0) function.

KEEPING IT SIMPLE

Random Rectangles

The following program and its accompanying description are designed to make the new user/programmer feel more comfortable with the PLOT statement. This and other statements are fully described in our easy-to-understand Programming and Reference Manual, available through your dealer or direct from Compucolor Corporation. The Manual details the features of the machine and of the programming language BASIC. It starts out at an introductory level and progresses to sophisticated explanations of machine capabilities. It makes a handy reference, and when combined with study from one of the books recommended in the Instruction Manual, it will have the new user programming in no time at all.

The program below will draw randomly-sized rectangles of random color on the screen. The program is quite simple and should be easy to follow with but a little concentration.

The first step is the defining of a function that allows us to specify a range and have random numbers generated in that range (line 20). After having set the background color to black and cleared the screen (line 110) a loop is created with a FOR statement that will cause the program to repeat the drawing of rectangles a random number of times from 6 to 12 (line 210). Note the use of the user-defined function to select the number between 6 and 12.

The next step of the program is the random selection of the rectangle color. The colors have PLOT numbers from 0 to 7, but since a black rectangle would not show up on the pre-set black background, our random number function is set to choose only numbers from 1 to 7.

After the screen is prepared and the color set, the points that define the rectangle itself must be determined. As you can see in Figure 1 below, the four required coordinates can be derived from only 2 points, (X1,Y1) and (X2,Y2). Since the COMPUCOLOR II has 128 plot positions (from 0 to 127), the random function is set at 128 to determine coordinates for the two key points (lines 260, 270, 280, 290).

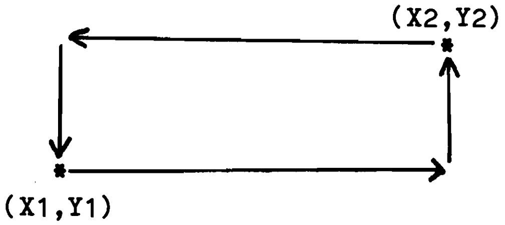


FIGURE 1

A subroutine is used to actually draw each rectangle on the screen. It is entered through line 300 with a GOSUB statement. The subroutine itself consists of entering the Point Plot mode by the PLOT 2 statement, and drawing the first point specified. The graphic Plot Submode 242, or Vector mode, is then used to draw the four sides of the rectangle. In this mode the computer draws a line from the last point plotted to the next specified point. Before using the mode, it is necessary to draw the first point in Point Plot mode (line 1000) to give the machine a starting point from which to draw the vector. After the initial point and the four vectors are drawn, we escape from plot mode by the PLOT 255 command (line 1030) and control is then passed back to the main body of the program at line 400. This NEXT statement causes the program to go back and repeat the process in order to draw another of the 6 to 12 rectangles specified. When the program falls through the loop, that is, when all of the rectangles specified by the random number function in line 210 have been drawn, the program will be at line 500. This line has no actual computational function; it is inserted simply to cause a slight delay in the program before it reaches line 510 and repeats the entire process of screen clearing, choosing the number of rectangles, color, etc. Try entering the program into your COMPUCOLOR II and see if you get the results you expect.

```
10 REM DRAW RANDOM RECTANGLES
20 DEF FNR(X) = INT(X*RND(1))
100 REM SET COLOR AND ERASE PAGE
110 PLOT 6,0,12
200 REM DRAW 6 TO 12 RANDOM RECTANGLES
210 FOR I=1 TO 6+FNR(7)
220 REM SET VISIBLE FOREGROUND COLOR
230 PLOT 6,1+FNR(7)
250 REM COMPUTE RANDOM COORDINATES FOR RECTANGLE
260 X1 = FNR(128)
270 X2 = FNR(128)
280 Y1 = FNR(128)
290 Y2 = FNR(128)
300 GOSUB 1000:REM DRAW
400 NEXT I
500 FOR I=1 TO 1000: NEXT I
510 GOTO 100

1000 PLOT 2,X1,Y1
1010 PLOT 242,X2,Y1, X2,Y2, X1,Y2, X1,Y1
1020 PLOT 255
1030 RETURN
```

Notice that a little planning ahead has gone a long way here. The simple function created in line 20 is used a total of 6 times and saves a lot of steps. The program can be easily changed by specifying different ranges for color (line 230), number of rectangles (line 210), and size of rectangles (lines 260-290). You might try some variations. Remember that typing LINEFEED (↓) will halt program execution and return you to BASIC, and the program will still be in memory.

SOFTWARE EXPLAINED

Star Trek Strategy

The game Star Trek has become very popular, and the version created for the COMPUCOLOR II is among the most exciting ever played. The colors and graphics make it come alive for you, the user, Captain of the Enterprise. The game is easy to play and has a simple strategy, but because it is quite unlike any board games you might have played, it does take some getting used to. The first information you are asked to supply is the Star Date, and as you play several games you will see that the degree of dificulty of the game you get is a result of the Star Date you select. So, if you play a game and get destroyed, but you think you have a new strategy, just enter the star date of that game, and the same number of Klingons, Star Bases, and stars will be allotted at the start of the game. Some players have favorite games that they play over and over. If you really like a challenge--try Star Date 5!

The Galaxy is divided into an 8 by 8 quadrant grid which is in turn divided into an 8 by 8 sector grid. The position of the Enterprise is graphically shown on the short-range sensors, and its coordinates appear in large yellow characters in the screen's upper left corner.

The long-range sensors display the position of the Enterprise in a blinking number. The other numbers on the sensors give information about stars, (the number of which in any quadrant is displayed in yellow) Klingons, (displayed in red) and star bases (number of stars displayed in green). The contents of all adjacent quadrants are also visible, and as the Enterprise moves throughout the galaxy, the quadrants that come into range are displayed on the sensor scan. Eventually, this gives a clear picture of the galaxy, and you know the exact location of all Klingons, stars, and Star Bases.

It is important, as you have no doubt discovered during play, to learn the location of any starbases as soon as possible in the game. When the Enterprise comes within 1 sector of a Star Base, the shields are dropped, the Condition is "Docked", and all energy is fully restored. For this reason, many players prefer to avoid Klingons completely until a Starbase is located and they have a ready source of energy.

You will notice that when the game first starts, the status (appearing at the right of the screen), is CONDITION RED. This danger signal is given because there is no energy in the shields, and any Klingon attack would destroy the Enterprise. The first move of every game should be to put some units of energy in the shields. You will discover by trial and error the amount of energy needed to withstand the usual Klingon attack. Putting too much energy in the shields will severely limit your mobility, while putting in too little will leave you vulnerable to Klingon attack. Once the shields are prepped, you are ready to travel the universe in search of Klingons. Your ultimate goal, of course, is to rid the Galaxy of the evil Klingons, and control the area for the Federation.

The commands listed in the lower right of the screen are all that are needed to play the game. They are:

0-Warp Engine Control

Course is in a circular numerical vector arrangement as shown in Figure 2. Integer and real values may be used. Therefore, course 1.5 is halfway between 1 and 2. A vector of 9 is automatically changed to 1. After choosing course, warp speed is selected. A warp factor of 1 is the size of one quadrant. Therefore, to get from quadrant 6,5 to 5,5 you would use course 3, warp factor 1.

1-Shield Control

Defines the number of energy units to be assigned to the shields. The energy is taken from the total ship's energy.

2-Phaser Control

Allows you to destroy the Klingons by hitting them with suitably large numbers of energy units to deplete their power. The phasers are wide-range weapons, and require no target calculations.

3-Photon Torpedo Control

Torpedos must be precisely aimed using the same course selection described in command 0. Note the fate of Klingons after a direct hit!

The movement in the Star Trek game must be carefully planned because it is confusing at first. You use the ship's computer to lay in your course, and even though the distance and direction are mathematically calculated, you must determine the course yourself, and unless you stay ahead of the computer you're liable to end up lost in space! The course is computed as shown below:

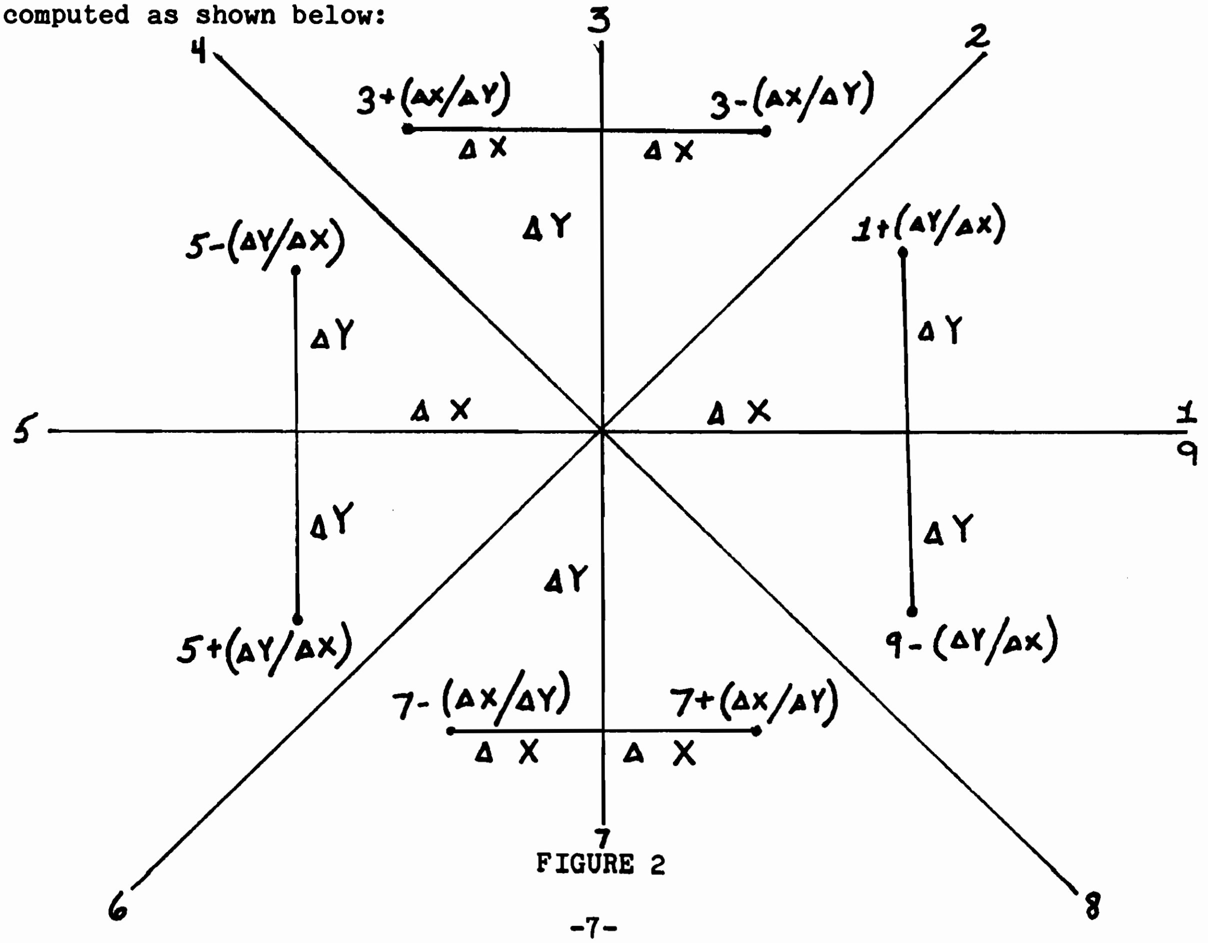


FIGURE 2

There are a few details about which you should know in order to become an effective starship Captain. The first is that the status of devices, as shown in the lower right of the screen, can greatly affect your odds in any particular game. If the sensors go down, the displays will not be updated, and Klingons can move into your quadrant unannounced. It is important to be very careful to navigate AROUND stars. Attempting to go through a star that is visible on your short-range sensor screen will stop the ship at the point where it encounters the star. If a Warp Engine shut down appears on the Device Status record, your speed will be severely curtailed. It will take the elapse of several plays for Scotty to fix the engines, but eventually they will again be operable.

Klingons do not have to remain in the quadrant where you first encounter them. Each time you issue a command, Klingons within a distance of 3 quadrants may converge on your quadrant up to a maximum of 9 Klingons. It will take some practice to become truly adept at this challenging game, but with a little cleverness and some patience you will soon be a skilled player.

USERS' SOFTWARE FILES

Software Exchange

ColorCue will provide the means through which readers may sell, trade, or contribute programs to Compucolor Corporation. The next issue will feature a complete explanation of buying policies and standards for acceptance.

****

COMING UP

Othello: COMPUCOLOR II vs. all challengers
User's Software Files: Software sales and acceptance policies
RS232-C Interface: Plugging in to the Big Machines
More PLOT's and some TIMEly Applications

****

We hope you have found this first issue of ColorCue illuminating. In order that we meet your needs and interests, write to us with questions, comments and criticisms. Keep on experimenting and let us know what you come up with! Write to:

COMPUCOLOR CORPORATION
P.O. BOX 569
NORCROSS, GEORGIA 30071

ATTN: Susan G. Sheridan

Vol. I.2 Nov.'78

COLORCUE

$1.00

READERS--

Certainly by this time you have thoroughly perused all of the material presented in October's issue. We are pleased to say that response has been quite good, with readers being especially appreciative of the Star Trek instructions. As a result of the articles in the last issue, many new users have seen the coming and going of thousands of Random Rectangles, while the home programmers are coming up with some innovative uses for the Scrolling Patch. Our explanation of Star Trek Strategy has undoubtedly contributed to the demise of a number of Klingons. If you still have not tried the Star Trek game, you can use the order blank attached to this month's ColorCue to send to us for your Star Trek SofDisk Album. It includes the Lunar Lander and Shoot games, as well as Tic-Tac-Toe.

****

This Issue's MENU:

****

This issue presents not only the usual articles and information, but also a feature on the use of the RS-232C interface. We know that many of you have access to large computers, and using the COMPUCOLOR II as a terminal is an important feature for you. Although the procedure is described in the Programming and Reference Manual, we have chosen this issue of ColorCue to present it to all users.

November ColorCue also includes a description of the trade and acceptance policies for the Users' Group software. We hope that by defining the policies, more of you will be encouraged to send us those creations on which you have been working so hard. We are actively soliciting software, so don't assume that we're not interested in your project. Send it to us and let us have a look!

ADVANCED APPLICATION

The Real Time Clock

The COMPUCOLOR II contains a Real Time Clock that gives the programmer an added dimension in creativity. It has many applications, and can be used for time testing in educational or game programs. The program below shows how get the Real Time Clock to display in the upper right hand corner of the screen. This is a very simple program, but because it gives the memory values necessary for the Real Time Clock, you will be able to elaborate on this program and create interesting effects.

The first lines (100 - 150) set up the program and define variables. The program allows you to enter any starting time that you chose. This is put into the proper place in memory by the POKE statements in lines 180, 190, and 200. The time is then retrieved by the PEEK statements in lines 230, 240, and 250. Lines 270 - 290 align the digits for a more easily read display. This is done by adding 100 to each time value, and then printing out only the last two digits. This assures that any leading zeroes will be displayed. Note the use of the string function STR$ to change the numeric value into a string variable. The PLOT and PRINT statements in line 300 define the area of the screen to be used for display. This PLOT statement can be changed to display the time readout in any portion of the screen. Line 330 makes sure that the display is not updated unless the time has changed. If this line were not included, the program would still work, but the display might be jittery because of the constant needless update.

```
100 REM *** TIME ***
110 PLOT 6,34,12,14
120 REM *** SET TIME ***
130 INPUT "DO YOU WANT TO SET THE TIME (Y OR N)? ";I$
140 IF I$ = "N" THEN 210
150 INPUT "ENTER TIME (HR, MIN, SEC) ";H,M,S
170 REM *** PUT CORRECT TIME INTO MEMORY ***
180 POKE 33211,H
190 POKE 33210,M
200 POKE 33209,S
210 REM *** PRINT TIME; GET TIME ***
230 S=PEEK (33209)
240 M=PEEK (33210)
250 H=PEEK (33211)
260 REM ** PUT 0 IN FRONT OF SINGLE DIGIT NUMBERS **
270 S$=RIGHT$ (STR$ (S+ 100), 2)
280 M$=RIGHT$ (STR$ (M+ 100), 2)
290 H$=RIGHT$ (STR$ (H+ 100), 2)
300 PLOT 3,50,0:PRINT H$;":";M$;":";S$
320 REM ** WAIT UNTIL SECONDS HAVE CHANGED
330 IF PEEK (33209)=S THEN 330
340 GOTO 230
```

The time is actually updated every 8/15 seconds, so that every few seconds, the time may seem to advance faster for one count. You may find that your COMPUCOLOR II gains or loses a few seconds per day. This can be corrected by adding this number of seconds once every day.

KEEPING IT SIMPLE

Circular Plots

Last month's Random Rectangles was an example of the use of the PLOT statement to draw a simple figure at random. This month we shall show a more involved screen display. The program below concentrates on the use of the graphics plot modes to draw various circles and ovals whose parameters you select.

The first plot statement sets the background color to blue, the foreground color to white, and clears the screen. Lines 20 and 30 define variables for future use. Line 130 sets the radius at 50 and line 140 places the center point of the circle in the center of the screen. These two statements assure that the plotting will not go off the screen. Lines 150 and 160 ask for two user inputs. The first asks the user to select the angles which will determine the screen display. The START and STOP values determine what portion of a circle will be drawn. The final value, STEP, selects the angle at which the lines will be drawn. The second input selects an aspect ratio. The aspect ratio is a screen character width to height ratio that can cause distortion to the circle appearance. The aspect ratio on the COMPUCOLOR II is 3 to 4, so if the user inputs .75 in response to line 160, the circle will appear balanced. Line 170 makes certain that no intolerable aspect ratios are accepted.

Lines 200 and 210 erase the screen and set plot mode, while line 220 sets up the loop that cause a complete circle to be printed. Lines 230 and 235 convert the degrees into radians and calculate the circle parameters. Line 240 causes a line to be drawn, and line 250 completes the FOR/NEXT loop that draws all the lines of the circle. Line 260 exits from plot mode, and the last line of the program sends control to the first portion of the program so that the user can choose another set of parameters.

```
10 PLOT 14,6,39,12
20 RD=3.14159265/180
30 P2=3.14159265*2
100 REM   PLOT LOOP
130 R = 50: REM CIRCLE RADIUS IS SET AT 50
140 XO=64: YO=64: REM CIRCLE CENTER IS SCREEN CENTER, 64,64
150 INPUT "ANGLES (0-360) START,STOP,STEP :  ";D1,D2,D3
160 INPUT "ASPECT RATIO  ( <1) : ";A
170 IF A> A THEN A=1/A
200 PLOT 12
210 PLOT 2
220 FOR D=D1 TO D2 STEP D3
230 RO = D*RD: REM RO= RADIANS CONVERTED FROM DEGREES
235 RO=RO-P2*INT (RO/P2)
240 PLOT XO + A*R* COS(RO), YO + R*SIN (RO), 242
250 NEXT D
260 PLOT 255
270 GOTO 100
```

The angles that you select will determine the shape of the circle that is plotted. To get an entire circle, you must enter 0 and 360 for the START and STOP points. Enetering 30 and 60 for those same points would plot only 30 degrees of the circle. Choosing a high step value will cause a rough-edged circle to be plotted, whereas a small value will result in smoother edges. You can try experimenting with various aspect ratios. Notice the effect that different aspect ratios have on circles defined by the same angles.

SOFTWARE EXPLAINED

Othello

When Gabriel Toys first released its new board game, Othello, in 1975, it was called "The first completely new game in a hundred years". Now, through the COMPUCOLOR II, Othello has been made newer than ever. CBS toys, makers of the Othello game, have issued the only license for a computer version of the game to Compucolor Corporation. Othello combines perfectly balanced elements of skill, chance, and excitement into a sensational game. Athough strategy and rules are simple, winning will require some forethought and presence of mind.

The COMPUCOLCOR II allows a lot of flexibility in Othello-playing. You may choose to play against another person, in which case the COMPUCOLOR II will referee the game, keeping score and disallowing illegal moves. You may play against the computer, or you may pick up some playing tips by watching the computer play itself.

Othello is played on a standard 8 x 8 square playing board as is used for chess or checkers. You will see that each game starts with four chips in the same configuration. Every time you move, you must outflank one or more of your opponent's chips. This will cause those chips to turn to your color. This is known as a flip (because in the board game, the chips have a different color on each side and must be manually flipped at every play.) You outflank an opponent's chip by placng your chip on a square so that the opponent's chip(s) are in between your new chip and some other previously played chip of your color. (See Figs.1&2) A chip may outflank

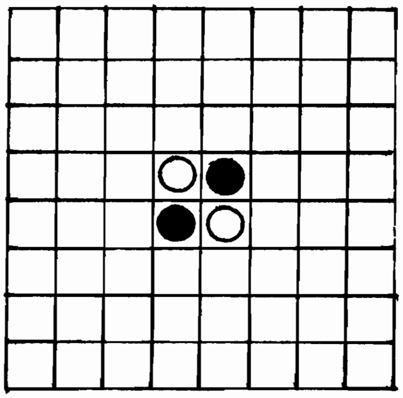


FIG 1

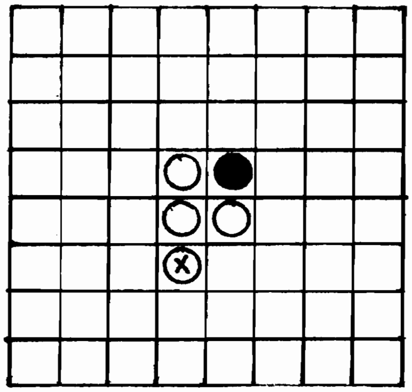


FIG 2

any number of opponent's chips in any one turn, in any of several directions -- horizontally, vertically, or diagonally. You must always move so that you outflank at least one of your opponents's chips. If you cannot make such a move, you must pass your turn. You instruct the computer to do so by entering a 0 at your turn. The game is over when neither player can make a legal move, and the player with the majority of squares occupied by chips of his color is the winner. As you play a number of games, you will discover that certain squares are very important to winning the game. All edge squares are desirable, because they prevent your being outflanked from one direction, and all corners are extremely important because they can never be outflanked. As you play, you should keep in mind the portions of the board that you wish to control and maneuver your playing so as to acquire these key positions. You will discover that positions can change drastically toward the end of the game, when rows are longer, and flips turn many more chips. Note Figs. 3&4.

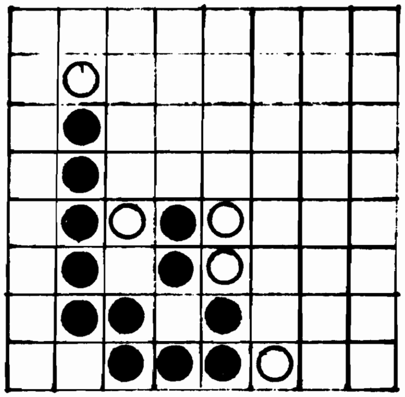


Fig 3

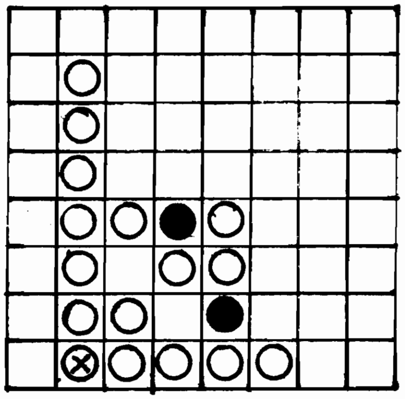


FIG 4

USERS' SOFTWARE FILES

Acceptance Policies

Part of the purpose of the ColorCue newsletter is to facilitate the exchange of software among users. Having access to a wide variety of programs can make the COMPUCOLOR II even more useful for you. Many of you are spending time creating programs that play games or help with finance or home management. We are interested in seeing these programs and in reviewing them for acceptance to the Software Files.

When a program is submitted to us, we examine it for originality, function, and style. It is very important that good documentation be received so that the program's purpose is easily discerned. You should submit programs on SofDisk, but if you have a printer, you might also send us the source listing. Programs will fall into three classes. The first are those programs which are judged so excellent that they will be sold on SofDisk Albums. For these programs, we will pay the originator anywhere from $100 to $1000, depending on program content, size, originality, etc. The second class of programs will be those that will become part of the current Users' Software Files. When a program is accepted to the Files, the programmer will receive a program of approximately equal length and complexity from among those already submitted to the Files. There will be several categories of program types, and the programmer may specify the category of his preference. Obviously, it will take some time to build the Files up to the point where users have a wide selection of materials. However, we are hopeful that with your cooperation, the files will build rapidly. Until there is substantial material from which to chose, programmers whose products are accepted will receive through the mail not only their SofDisk upon which the program was submitted, but also a pair of formatted COMPUCOLOR II SofDisks (a $20.00 value). Users who do not contribute to the files will be able to purchase the programs. A schedule of prices will be listed in future issues.

Submissions in the third category of programs, those not considered ready for release, will simply be mailed back to their originators. Whenever possible, we will try to include suggestions on how the program can be improved and made acceptable.

We hope that this brief introduction makes acceptance policies clear. Please send in any programs you have written so that we may consider them for publication. We are hoping that the Users' Software Files will become an important feature of the Users' Group.

THE RS-232C INTERFACE

Plugging Into Peripherals

The RS-232C interface allows the user to connect any RS-232C compatible device to the COMPUCOLOR II. This enables most serial printers to be interfaced without any addtional software.

In order to connect to another machine you must be sure that the COMPUCOLOR II baud rate is compatible with that of the other machine. The default baud rate of the serial port is 9600 baud with 1 stop bit. This rate can be changed by using the ESC R sequence. A7 ON and A7 OFF give 2 and 1 stop bits, respectively. The baud rate is selected by giving a sequence which includes the stop bit, ESC R, and some number from 1 to 7 chosen from the chart below.

BAUD RATE SELECTION

NUMBER KEY	1	2	3	4	5	6	7
BAUD RATE	110	150	300	1200	2400	4800	9600

After the baud rate has been properly set, data can be transmitted to the RS-232C serial port by executing an ESC M sequence. As listed in the Instruction Manual, this ESCAPE sequence sends all inputs to the COMPUCOLOR II (including all keyboard inputs and BASIC outputs) to the RS-232C port instead of to the CRT screen. Thus, the only way to break out of this mode from the CRT mode is via the CPU RESET key or in Disk BASIC by executing a POKE 33265,0 statement which resets the BASIC Output Flag to send characters to the screen in visible cursor mode.

The COMPUCOLOR II can be used as a data communications terminal with the RS-232C interface. The user enters terminal mode by stiking CPU RESET. Full or Half duplex can be set with the ESC F and ESC H codes, respectively. In half duplex, all keyboard inputs are echoed to both the

CRT and the RS-232C port. In full duplex mode, all keyboard inputs are directed only to the RS-232C port. Your choice of one or the other depends upon the requirements of the machine to which you are interfacing.

When interfacing a modem or acoustic coupler to the RS-232C serial port, it usually is not necessary to switch the transmit and receive data lines because the RS-232C port is configured as if the COMPUCOLOR II were a data communications terminal. On some computer systems that use 7-bit ASCII codes, correct parity is required. The COMPUCOLOR II is set up to transmit and receive 8 bit ASCII characters and data which precludes parity checking. The COMPUCOLOR II can receive any form of parity from the host system as long as it is in the normal text mode (non-plot modes). But it transmits with the parity bit always low. The host system should not send the following control codes unless it can control the state of the parity bit : plot mode (02), cursor XY lead-in (3) and CCI lead-in (6). In this case, proper communication will require a special communications program that executes in the user's RAM workspace, or else the host computer system can be configured to receive ASCII with parity bit low.

Another problem that may be encountered is that the COMPUCOLOR II responds to almost every ASCII control code and escape sequence. If the host computer sends control codes other than NULL, CARRIAGE RETURN, LINEFEED, and ERASE PAGE, then the COMPUCOLOR II may respond in an unexpected fashion. In these cases it is probably best if the host computer treats the COMPUCOLOR II as a TELETYPE instead of a CRT display terminal.

READING THE KEYBOARD

We have received a number of requests from COMPUCOLOR II users who want to make inputs to BASIC without the limitations imposed by the INPUT command. Our software people went to work and came up with the following BASIC routine which, when added to your program, will enable you to do any or all of the following:

a. Enter a character without its being echoed to the screen.
b. Enter a character without hitting RETURN.
c. Wait X seconds for a character and continue program if one is not hit.

```
100 REM **** GET CHARACTER FROM KEYBOARD ****
110 DATA 245, 229, 197, 1, 206, 40, 205, 36, 0, 202, -1, -1, 11
120 DATA 121, 176, 194, -1, -1, 29, 194, -1, -1, 17, 255, 255,195
130 DATA -1,-1, 95, 175, 87, 175, 50, 255, 129, 193, 225, 241,201
140 REM ** FIND END OF BASIC RAM **
150 TM = 256*PEEK(32941) + PEEK(32940)
170 REM  ** CHECK TO SEE IF IT'S ALREADY LOADED **
180 IF TM> 65503 THEN 250: REM ** IT'S NOT LOADED **
190 RESTORE 110
200 FOR I = 1 TO 39: READ A
210 IF A> = 0 AND A<> PEEK (TM +I) THEN I=39: A=999
220 NEXT I
230 IF A< 256 THEN 390: REM ** IT'S LOADED, JUST SET JUMP
240 REM ** COME HERE TO LOAD PROGRAM **
```

```
250 TM = TM-39: RESTORE 110
260 REM ** LOAD MACHINE INSTRUCTIONS **
270 FOR I = 1 TO 39:READ A: ROKE TM + I, A-(A<0): NEXT I
280 REM ** LOAD ADDRESSES **
290 Z = TM +29:AD=TM+11:GOSUB 350
300 Z = TM +7: AD=TM+17:GOSUB 350
310 Z = TM +4: AD=TM+21:GOSUB 350
320 Z = TM +32:AD=TM+27:GOSUB 350
330 GOTO 370
340 REM ** LOAD ADDRESS Z AT ADDRESSES AD, AD+1 **
350 ZZ = INT(Z/256): POKE AD,Z-256* ZZ:POKE AD +1, ZZ:RETURN
360 REM ** CHANGE END OF BASIC RAM **
370 Z = TM:AD = 32940:GOSUB 350
380 REM ** LOAD BASIC JUMP LOCATION **
390 Z = TM +1:AD=33283:GOSUB 350
400 POKE 33282, 195:REM ** LOAD JUMP
410 CLEAR 25: REM ** YOU'RE DONE **
420 REM ** YOUR PROGRAM STARTS HERE **
```

This routine must be executed prior to any part of your program. You can set the "CLEAR" in line 410 to meet your program's requirements. Make sure that no variables are dimensioned prior to this or they will be lost. Once this routine is executed, the BASIC "CALL" function can be used to accept keyboard inputs. The argument of the call is the number of seconds the program will pause for a character to be input. An argument of zero (0) will result in the longest delay -- 255 seconds. A return to BASIC occurs when either a key is hit or the time expires, with the value of the call set to the ASCII value of the character hit. If no character is hit, then -1 is returned at the expiration of time.

The following examples demonstrate how this routine might be used.

a. This routine will pause 5 seconds to wait for a character. If a character is hit, it is printed.

```
500 Y = CALL (5)
510 Y<> -1 THEN PRINT CHR$(Y)
```

b. This routine fills the string A$ with characters until "RETURN" is hit. The characters are not printed on the screen. Notice that a "RETURN" (ASCII 13) is treated just like any other charcter by the routine. If no check was made for it, it would be imbedded in the string, and you could not exit the loop.

```
500 Y=CALL(0): IF Y=-1 THEN 500: REM ** DON'T LET CALL TIME OUT **
510 IF Y=13 THEN 540: IF RETURN IS HIT -- YOU'RE DONE
520 A$ = A$ + CHR$(Y): REM ADD CHARACTER TO STRING
530 GOTO 500: REM GO GET NEXT CHARACTER
540 REM ** CONTINUE PROGRAM
```

c. This routine will wait 2 seconds for a "Y" (ASCII 89) or "N" (ASCII 78) to be entered. It prints "YES" if a "Y" is entered, and "NO" if a "N" is entered.

```
500 Y = CALL (2)
```

```
510 IF Y=89 THEN PRINT "YES"
520 IF Y=78 THEN PRINT "NO"
```

COMING UP

String Functions
Menu
Checkbook
INPUT table values and functions
INPUT flag types

Note: Those of you with older versions of the SHOOT game may notice a mistake in the instructions. The phrase "70 0 9:PLOT 10" should read "70 (enter) 9(enter)" Excuse us!

Please address all inquiries and suggestions to: Susan G. Sheridan
Compucolor Corporation
P.O. Box 569
Norcross, GA 30071

VOL I.3 DEC 78 COLORCUE $1.00

READERS--

Letters are now beginning to come in on a fairly regular basis, and we will be publishing some of them so that you can compare thoughts. We have had requests for various pieces of information, and though we may not be able to cover every question asked, we will try to answer those most frequently mentioned. November's issue was unpardonably late, but contained some useful information. Because of limited space, it is not possible for us to cover every aspect of each subject in ColorCue. Readers may still have some questions about subjects in some of the articles we have published. Address specific questions to us, and perhaps in another issue we will cover other aspects of these features.

****

This Issue's MENU

****

On the Sof-Disk front -- We have three new Albums ready for shipment. These are BONDS, EQUITY, and INCOME TAX '78. The BONDS Album will be of use to those of you who want to make more informed investment decisions. You can determine yeild and maturity accurately and quickly. The EQUITY Sof-Disk can figure total long-range costs to aid in purchasing and depreciation decisions. The BONDS and EQUITY Sof-Disk Albums sell for $19.95 each, and the TAX '78 Album is $29.95. All three include printed instructions to make program use easier. All of these progams require 16K of memory. We are pleased to announce that our INCOME TAX'78 program is ready just in time to caluculate your 1978 return. It is very easy to use, and it calculates the form 1040 and Schedule A. The program is set up so that all of the entry lines are numbered the same as these forms. The lines are color-coded so that you know exactly where to enter a number or when to wait for a calculation. When all your entries are finished, you will have a fast, flawless, return. We can't promise that doing your taxes will be fun, but the TAX '78 program will make it less of a strain. Order from your dealer, or use the enclosed form.

Those of you who have not yet purchased your Programming and Reference Manual may want to do so soon. Many dealers have them in stock. It explains the special features of BASIC on the COMPUCOLOR II and is indispensable for using color and graphics.

ADVANCED APPLICATION
Input Flag Values and The INPUT Table

On the COMPUCOLOR II, INPUT flags make the world go round. Any time a character is passed from one device to a program, or from a program to a device, or from a program to another program, an input flag is associated with it and the value of this flag determines the character's destination and usage. Characters from the keyboard come in on KEYFL (33247), Output from BASIC goes out on BASFL (33265), and output from FCS on LOFL (33273). (Input to FCS is echoed on another flag, FCSFL (33249), so that output from commands entered at the keyboard may be sent to the serial port. See Escape G.)

When a character is generated from some source in the COMPUCOLOR II, the value of the flag associated with it is used as an index into the INPUT table. As each entry is an address of a routine which will be called with the character and the address of the flag as parameters. The routine can then modify the flag value, setting a new destination for the next character, store or display the character, or both store that character and modify the flag.

As a common example let us examine the CRT mode. When the keyboard (or BASIC or FCS) is in CRT mode, the flag associated with it contains the value 0. The entry of the INPUT table corresponding to zero is the video display routine. When the disply routine receives a character it either stores the character in the display buffer if the character is printable, or it interprets it as a control character if it is not. Control characters are used to erase the screen or a portion thereof and otherwise manipulate the display. When, for example, control-C (value 3) is sent to the CRT, the display routine alters the value of the current flag from 0 to 3 and returns. This means that the next character to come in on that flag will not go to the display routine and will instead be received by the routine corresponding to the value three in the INPUT table. Therefore the next character will not be displayed even if it is printable. When a character is received by the routine at entry 3, the character is stored as the new cursor X value and the flag is changed from value 3 to 5. The routine then returns to the caller. When a character is received by the routine at entry 5, the character is stored as the new cursor Y value and the cursor is moved to its new location. The flag is then changed from value 5 to 0 and the next character will be interpreted by the display routine. Thus we can see why control characters do not affect the display when used as data for a previously initiated sequence.

All multi-byte control sequences are implemented in this way. Graphics mode is a special case where the flag of the generating device or program is set to 2 and the usage of the character is determined by the system graphics mode.

When BASIC is entered from the CRT mode the BASIC interpreter sets the flag value to 23, which entry in the INPUT table is the address of BASIC's input control routine. Therafter all data entered goes directly to BASIC without being displayed. BASIC then may display or interpret the data as it chooses. In this manner BASIC echoes the keyword 'SAVE' whenever control-shift S is entered. BASIC's output flag, BASFL, may be controlled from the program by using PLOT statements, but it should be emphasized that the Keyboard flag, KEYFL, will remain set to 23 for as long as BASIC requires input. (When BASIC does not require input, the flag is set to 0, which causes data to be displayed.) A similar mechanism controls input to FCS.

More information concerning the use of INPUT flags will be forthcoming future issues of ColorCue.

ENTRIES OF THE INPUT TABLE

00	System Display Routine
01	Blind Cursor Mode Display Routine
02	Graphics Mode
03	Expecting Cursor X Address (changes flag to 5)
04	Color Pad Plot Mode (non-color control characters go to Display)
05	Expecting Cursor Y Address (changes flag to 0)
06	Expecting New CCI Code (changes flag to 0)
07	Expecting Blind Cursor X Address (changes flag to 8)
08	Expecting Blind Cursor Y Address (changes flag to 9)
09	Expecting Blind CCI Code (changes flag to 1)
10	Unused
11	Unused
12	Garbage Dump (all characters ignored; flag not chaged)
13	FCS Input Routine
14	Serial Port Output
15	Unused
16	Unused
17	Unused
18	Expecting Baud Rate 0 - 7 (changes flag to 0)
19	Unused
20	Unused
21	Unused
22	Unused
23	BASIC Input Routine
24	Unused
25	Expecting Fill Screen Character (changes flag to 0)
26	Unused
27	Expecting Escape Sequence (changes flag to 0,4,13,18,23,25)
28	Unused
29	Unused
30	Unused
31	Unused

Unused INPUT table entries all jump to location 81C5 hex (33221 DEC) and may be used by the programmer if an appropriate jump is poked into that location.

KEEPING IT SIMPLE
Character String Manipulations

The COMPUCOLOR II Extended Disk BASIC has three functions which are used to manipulate character strings. They are:

```
LEFT$(X$,I)
RIGHT$(X$,I)
MID$(X$,J,I)
```

In each case, X$ represents the string being operated on, and 'I' represents the number of characters to be returned. The only differece between the functions is the location in the string where the 'I' characters will come from. With LEFT$, it will be the first 'I' characters, and with RIGHT$, the last 'I' characters. MID$ contains an additional variable 'J' which indicates the starting position within X$.

MID$ is the most flexible of these three functions and can in fact be used in place of the other two functions. For example, MID$(X$,1,I) is exactly equivalent to LEFT(X$,I). To use MID$ in place of RIGHT$, however, we must know the length of X$. The equivalency would therefore be MID$(X$,LEN(X$)-I+1,I). Obviously, the RIGHT$ function would be easier to use.

```
100 X$="ABCDEFGHIJKL"
110 PRINT LEFT$(X$,4)
120 PRINT MID$(X$,1,4)
130 PRINT RIGHT$(X$,4)
140 PRINT MID$(X$LEN(X$)-4+1,4)
150 PRINT MID$(X$,5,4)
```

Lines 110 and 120 will both print out the first four characters of X$ or 'ABCD'. Lines 130 and 140 will both print out the last four characters of X$ or 'IJKL'. Line 150 will print out the 4 characters starting with the 5th character or 'EFGH'.

If the number of characters (I) is greater than the number of characters available, the function will return what is available and will not generate an error. This frees the user from having to check the length of a string prior to using a function. A CF error (Call Funtion) will be generated, however, if 'I' or 'J' is less than one (1). In the above examples we change line 100 to X$="" (a null string) a CF error will occur in line 140. This is because the valve for J is evaluated as -3, an illegal value.

A problem frequently encountered is the need to right justify columns of numbers. The following routine will easily accomplish this where X is an integer number.

```
100 INPUT "X = ";X
110 X$ = "        " + STR$(X)
120 PRINT RIGHT$(X$,8)
```

This program converts X to a character string and add it to a string of blanks. By printing out the right portion of this string, all values will be right justified. A similar routine could be written which would also handle decimal numbers.

SOFTWARE EXPLAINED
The Deluxe Keyboard

This may well come more under the heading of 'Hardware Explained', but since the special function keys aid in the creation of displays, we will leave it in this category.

We have had a number of questions concerning the use of the special function keys, both from people making buying decisions and from people who already own the deluxe keyboard. The special function keys do not expand the capabilities of the machine, but like the numeric and color pads, they enable the user to accomplish a task with fewer keystrokes.

The special function keys are used in the CRT mode to create displays. These displays can be saved by entering the following BASIC program before the display is created:

```
0 PLOT 27,4:PRINT "SAVE SCREEN.DSP 7000 1000":END
```

You then hit CPU reset to return to the CRT mode, where the display can be entered. When the display is complete, hit ESC E to reset BASIC and RUN the previously entered one-line program. The ESC E sequence will produce the message 'READY', which will be saved along with the display. You may want to set the foreground and background colors to the same color as your display background so that this message will be invisible when saved.

Essentially, the special function keys send an ASCII character to the screen. The key at the far left labelled 'F0 VECT INC' sends ASCII 240, and the keys to the right each send a code one larger than the one to their left. The keys are labelled according to the vector graphics as listed in Chapter 9 of the Programming and Reference Manual. Therefore, these keys can be used to create any graphic disply that is possible on the COMPUCOLOR II. To give values for the various plot modes, the computer reads the ASCII values of the keys. (You will find these values in the back of the Programming and Reference Manual.) Try the following example:

```
CTRL B
CTRL @
CTRL @
F2
CTRL @
DEL
DEL
DEL
CTRL @
CTRL @
CTRL @
F15
```

It will draw a rectangle around the screen.

Corrections and Comments

November ColorCue contained a few unfortunate errors. They were small, but those of you who are experimenting with some programming know that even small errors can ruin the whole program. On page 2, line 140 should read IF I$="N" THEN 210 There was no line 220 in the program to which it could tansfer. Probably most of you were able to notice this by yourselves. On page 3, lines 20 and 30 are almost the same. Line 20 should read RD=3.14159265/180 and line 30 should read P2=3.14159265*2.

A reader from New York who is working on teaching himself BASIC says that he saves short example programs of various routines and functions, and then refers to them when writing programs. Some of you might find this clever idea quite useful.

Another reader has written in to ask if there is any way to make the screen scroll down instead of up. Sorry, but this is a complicated task. It involves changing the machine software. However, you may change the pertinent statements in the SCROLL program to make certain sections of your screen scroll down instead of up.

With this issue, you should all have recieved three issues of ColorCue -- October, November and December. If you are missing one or more of these issues, please let us know so that we can have it sent to you.

Write to:

Compucolor Corporation
P.O. Box 569
Norcross, GA 30071

ATTN: Susan G. Sheridan

Vol II.1 JAN'79 COLORCUE $1.00

COMPUCOLOR CORPORATION ANNOUNCES MAJOR PRICE CUTS

Readers --

Our big news this month is that prices have been reduced on a number of options and accessories. We are able to make these price cuts because of reduced memory costs, lower keyboard costs, and high sales volume. These price changes will make some of the most popular accessories more easily affordable. Those of you with Models 3 or 4 will note on the enclosed order blank that the price of the additional 16K of RAM has been cut in half. You can expand your machine's capabilites at a new low price. This is especially helpful for those of you with the Model 3 COMPUCOLOR II, since you can now update your machine just in time to figure your taxes with the new 16K TAX '78 Sof-Disk Album. The price cut that will undoubtedly be the most well-received is the one involving our formatted blank Sof-Disks. The price has been reduced from 2 for $20.00 to 2 for $9.95 -- a reduction of over 50%! We hope this will encourage you to come up with some clever programs.

****

This Issue's MENU

****

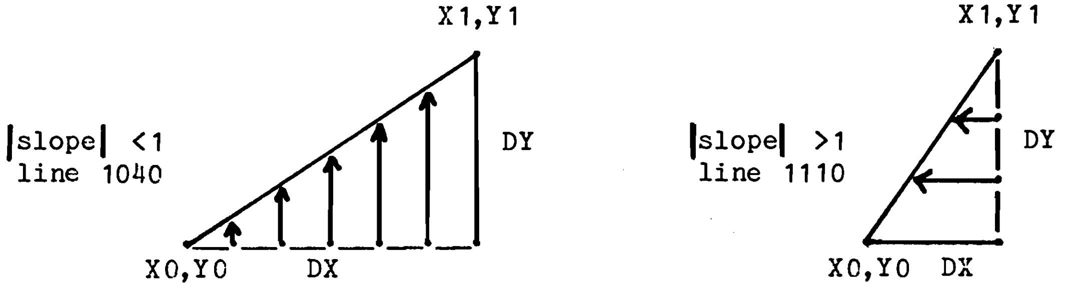
We know that some of you are about ready to make a contribution to the Users Group Software Files. We've had several promises from readers who will be sending in their best efforts. It looks like the Users Group is going to have a good selection of software for trade in just a few months, so think about sending in your programs for review. The Software Files will be a great way to find out how others are making use of their COMPUCOLOR II's. From our market surveys, it seems that most of you are interested in games of skill and financial applications. (What a sophisticated group!) So keep that in mind when you decide where to concentrate your efforts. Some of our Californian friends are asking about real time Space War games. We hope someome out there will come up with a spectacular one. Remember that an exceptional program can earn you up to $1000. Due to our mail volume, we will probably be starting a 'Readers Talk Back' sort of column next month. Some of you are prolific correspondents, but others of you are silent! All letters are welcome, so take a few minutes and let us know what you're thinking.

KEEPING IT SIMPLE
Dotted Lines

This month's topic was suggested by Barbara Freistuhler, of Los Angeles, California, who asked for a way to draw a dotted line between two points instead of the solid line drawn by the vector plot mode. Because the COMPUCOLOR II does not have infinite resolution, sloping lines can sometimes be irregular in appearance. Using a dotted line can create a neater, more concise display.

The program below allows you to specify two endpoints and an increment factor. The following explanation will illustrate the development of a simple but useful algorithm and the resulting program.

The idea behind the program is shown below. DX is the distance of the base line and DY the length of the side line formed by drawing vectors from the endpoints of a sloping line. Points are selected on the longest base line, and the slope determines where to plot the corresponding point on the line to be drawn.



This idea translates into the usable program below:

```
5 PLOT 12
10 INPUT "X0,Y0,X1,Y1,INC";X0,Y0,X1,Y1,INC
20 GOSUB 1000
30 INPUT "AGAIN?";A$
40 GOTO 5

1000 DX = X1 - X0: DY = Y1-Y0: PLOT 2
1010 IF DX=0 AND DY=0 THEN 1200
1020 IF ABS (DY) > ABS (DX) THEN 1100
1030 SL = DY/DX
1040 FOR X=X0 TO X1 STEP INC * SGN (DX)
1050 PLOT X, Y0+(X-X0) * SL+.5
1060 NEXT X
1070 GOTO 1200
1100 SL =DX/DY
1110 FOR Y=Y0 TO Y1 STEP INC * SGN(DY)
1120 PLOT X0+(Y-Y0) * SL+.5, Y
1130 NEXT Y
1200 PLOT X1,Y1,255:RETURN
```

Line 5 erases the screen. Line 10 uses the INPUT statement to let the user define the line's endpoints. Line 20 sends the program to the subroutine that calculates and plots the dots. Line 1000 calculates the length of the two lines that form the base and sides of the triangle pictured above. PLOT 2 enters plot mode. Line 1010 tests for the two endpoints being identical. If this is the case, control passes to line

1200, and only one point is plotted. Line 1020 selects the triangle base line from which to project the dots. If the triangle's side is longer than its base (i.e,|slope| > 1 ,) calculations are made at line 1100. Line 1030 calculates the slope by the 2-point method. (You recall this formula from Algebra I!) Line 1040 begins the FOR loop that actually draws the points, and line 1060 closes this loop. Line 1100 begins the calculation and plotting sequence for lines with the absolute value of the slope being >1. The program always plots both endpoints of the line. When the plot mode is exited and control is passed back to the main body of the subroutine, line 30 asks if another run is desired. Notice that this inquiry serves no function other than to delay erasing the screen to give you time to examine the dotted line. The program could easily be changed to terminate if the response to the question were 'NO'.

This formula for dotted lines can be incorporated into a program with the proper endpoints and increment value to acheive any desired display.

SOFTWARE EXPLAINED

Hangman Modifications

The market surveys tell us that many of you are involved with education in some way -- getting it or giving it -- at primary, secondary, or University levels. Some teachers have mentioned that being able to modify the Hangman program to insert a special word list would be useful. You can use words of your choice for vocabulary and spelling drills. Others can modify word lists to add a few favorite puzzlers. This month we'll show you how to select any words you choose and enter them into Hangman's lexicon.

The program below allows you to create a new LIB (library) file for HANGMAN. Enter the program as it appears below, and save it on disk for convenient access. When you RUN the program, it will guide you through the required word entry.

```
90 REM **************************************************************
91 REM
92 REM PROGRAM FOR CREATING USER DESIGNED FILE(S) FOR HANGMAN
93 REM
94 REM **************************************************************
95 REM
96 REM          THIS PROGRAM IS SET UP FOR WORDS NO LONGER
97 REM                THAN 16 CHARACTERS IN LENGTH
98 REM
99 REM **************************************************************
100 REM -- TO CREATE OR UPDATE A FILE
110 INPUT "UPDATE OLD LIBRARY (0) OR CREATE NEW LIBRARY (1) ?";A
120 IF A<>0 AND A<>1 THEN PLOT 28,11:GOTO 110
130 ON A+1 GOTO 210,140
140 INPUT "NEW LIBRARY NAME?";L$
150 INPUT "WORDS IN FILE?";W
160 FILE "N",L$+ ".LIB",W,16,8
170 FILE "R",1,L$+ ".LIB",1
180 PRINT "FILE ";L$;".LIB CREATED FOR ";W;"WORDS."
190 PUT 1,1;1,W
```

```
200 GOTO 250
210 INPUT "OLD LIBRARY NAME? ";L$
220 FILE "R",1,L$+".LIB",1
230 GET 1,1;N,W
240 PRINT "FILE ";L$;".LIB CONTAINS";N-1;"OUT OF";W;" WORDS."
250 INPUT "ENTER NEW WORD OR '0' ? ";W$
260 IF W$="0" THEN 300
270 N=N+1
280 PUT 1,N;W$[16]
290 GOTO 250
300 INPUT "END RUN (0), OR LIST WORDS (1) ? ";A
310 IF A<> 1 AND A<>0 THEN PLOT 28,11:GOTO 300
320 ON A+1 GOTO 390, 330
330 IF N=1 THEN PRINT "NO WORDS ON FILE.": GOTO 300
340 FOR I=2 TO N
350 GET 1,I;W$[16]
360 PRINT W$
370 NEXT I
380 GOTO 300
390 PUT 1,1;N,W
400 FILE "C",1
410 END
```

After you use this program, you will notice that a new file appears on the directory; for example, WORD.LIB. In order for the Hangman program to access this new file, a few lines in the Hangman program must be changed, as described below:

```
667 GOSUB 4000
668 (delete)
```

add:

```
4000 REM
4010 IF N8$="E" THEN FILE "R",1,"EASY.LIB",1
4020 IF N8$="H" THEN FILE "R",1,"HARD.LIB",1
4999 RETURN
```

For every new .LIB file you create, you must add a line to the subroutine that allows for access. For example:

```
4030 IF N8$="W" THEN FILE "R",1,"WORD.LIB",1
```

will allow the program to access a new .LIB file -- WORD.

DISK DUP

On some of the MATHTUTOR Sof-Disks there is a program called DUP. The MENU program does not access it, but many of you have found it on the directory and made use of it. It allows you to duplicate a Sof-Disk without having a second disk drive. This is very handy, as it's always a good policy to have a back-up copy of anything you're working on. For those of you who do not have a copy of DUP, a source listing is printed below. Just enter it into the COMPUCOLOR II and save it on a disk. While it runs, it gives you instructions on how to use it.

```
0 REM
1 REM ****************** DUP.BAS;01 ******************
2 REM
5 CLEAR 100
10 T = PEEK(32940) + 256 * PEEK(32941)
30 P = PEEK(32984) + 256 * PEEK(32985)
50 A = INT ((P+200)/256): B = (P+200) -A * 256
70 C = INT (T/256): D = T -256 * C
100 A = A+1: B = 0
102 C = C-2: D = 255
105 P = A*256+23: REM P NOW POINTS TO FIRST ATTRIBUTE BYTE
110 X = A: GOSUB 900: A$ = H$
120 X = B: GOSUB 900: A$ = A$+H$
125 A$= A$+"-"
130 X = C: GOSUB 900: A$ = A$+H$
140 X = D: GOSUB 900: A$ = A$+H$
160 I = 0
165 PLOT 6,2: INPUT "SOURCE DISK";Z: PLOT 27,4
166 X = I: GOSUB 900
170 PRINT "READ ";H$;" ";A$
172 PLOT 27,27: IF I=0 THEN GOSUB 800
173 PLOT 6,6: INPUT "DESTINATION DISK";Z: PLOT 27,4
174 PRINT "WRITE ";H$;" ";A$
190 I = I+INT ((T-P)/128)
194 IF I=>Q THEN 1000
195 IF I+INT ((T-P)/128) > 400 THEN I = 400-INT ((T-P)/128)
197 PLOT 27,27
200 GOTO 165
500 GOTO 1000

800 REM
810 INPUT "ARE THERE ANY ITEMS TO BE COPIED NOT SHOWN IN
    THE DIRECTORY?  ";Q$
820 IF LEFT$ (Q$,1) = "Y" THEN Q=400: RETURN
830 Q = PEEK(P+11) +256 * PEEK(P+12)
850 IF PEEK(P) = 1 THEN P = A * 256: RETURN
860 P = P+21
870 IF INT (P/128) = P/128 THEN P=P+2
880 GOTO 830
890 RETURN

900 REM X=VALUE TO BE HEXED
905 H$=" "
910 X1 = INT(X/16): X2 = X-16 * X1
920 IF X2<10 THEN X2=X2+48: GOTO 940
930 X2 = X2+55
940 IF X1<10 THEN X1=X1+48: GOTO 960
950 X1 = X1+55
960 H$ = H$ + CHR$(X1) + CHR$(X2)
965 IF X1<71 THEN RETURN
970 X1 = X1-71
980 H$ = " 1"
990 GOTO 940

1000 REM
1010 END
```

ADVANCED APPLICATION

The MENU Program

You may have noticed that the SAMPLER Sof-Disk Album has two ways of using the programs on that Album. You may access them individually, or you may type in 'ALL' and have the program cycle through a demonstration of each program. When 'ALL' is entered to the SAMPLER MENU, the program enters a demonstration mode where each program on the disk is run consecutively. When the last program is run, the program goes back to the first program and repeats the sequence. Since each program is loaded separately, one after the other, it may seem odd that the program knows to stay in the demonstration mode and not ask for user input. You may also wonder how the program is able to load and cycle through the programs in numerical order.

Actually, this is accomplished through a fairly simple procedure. It is done by using one byte of RAM which is outside the BASIC user memory. In this instance, location 32936 is used. If you look at the listing of MENU, you will see that when it RUNs, line 4 POKES a 0 (zero) into location 32936. Line 4000 POKES a 1 (one) into this address if 'ALL' is entered. If any other character(s) is entered, a zero is POKEd into this location.

When each program runs, it PEEKs at the value in 32935 to determine if it should run as a demonstration program (PEEK (32936) <> 0). (See BANDIT lines 126, 326, 465, 965 and 1040 as an example.) When the program returns to MENU, the statement:

```
LOAD"MENU":RUN 5
```

is used. This statement causes MENU to be loaded with execution starting at line 5. If this were not done, then line 4 would also be executed and the value in 32926 set back to zero. MENU checks the value in 32926. If the value is zero, the MENU program is executed in the normal way. If not zero, then the program is in the demonstration mode, in which case the value is incremented by one and the next program loaded.

NEW PRODUCTS

Keyboard and Pin Cushion

A horizontal pin cushion kit is available by writing to us. It is possible for you to install it yourself, but we do recommend that your dealer perform the installation. This kit will largely eliminate the side curvature of the screen display and make the display appear more rectangular.

A new keyboard case is avialable that replaces the old square-cornered model. The cost is $50.00, and it includes the up arrow cursor control key and switch, and the new Thin-Line case (avaiable in tan only). For installation, the board edge connector must be shortened by .25" to fit the new case. This is best done whith a fine file or fine grinder wheel.

COMING UP

Call for Articles
Reader Exchange
Personal Finance
Bio-Fix

Several readers have requested COMPUCOLOR II literature to give to friends or co-workers. We have included two of our new brochures with this issue.

When December ColorCue went to print, the last line of the first page was inadvertantly cut off. It should have read " and is indispensable for using color and graphics." Our apologies.

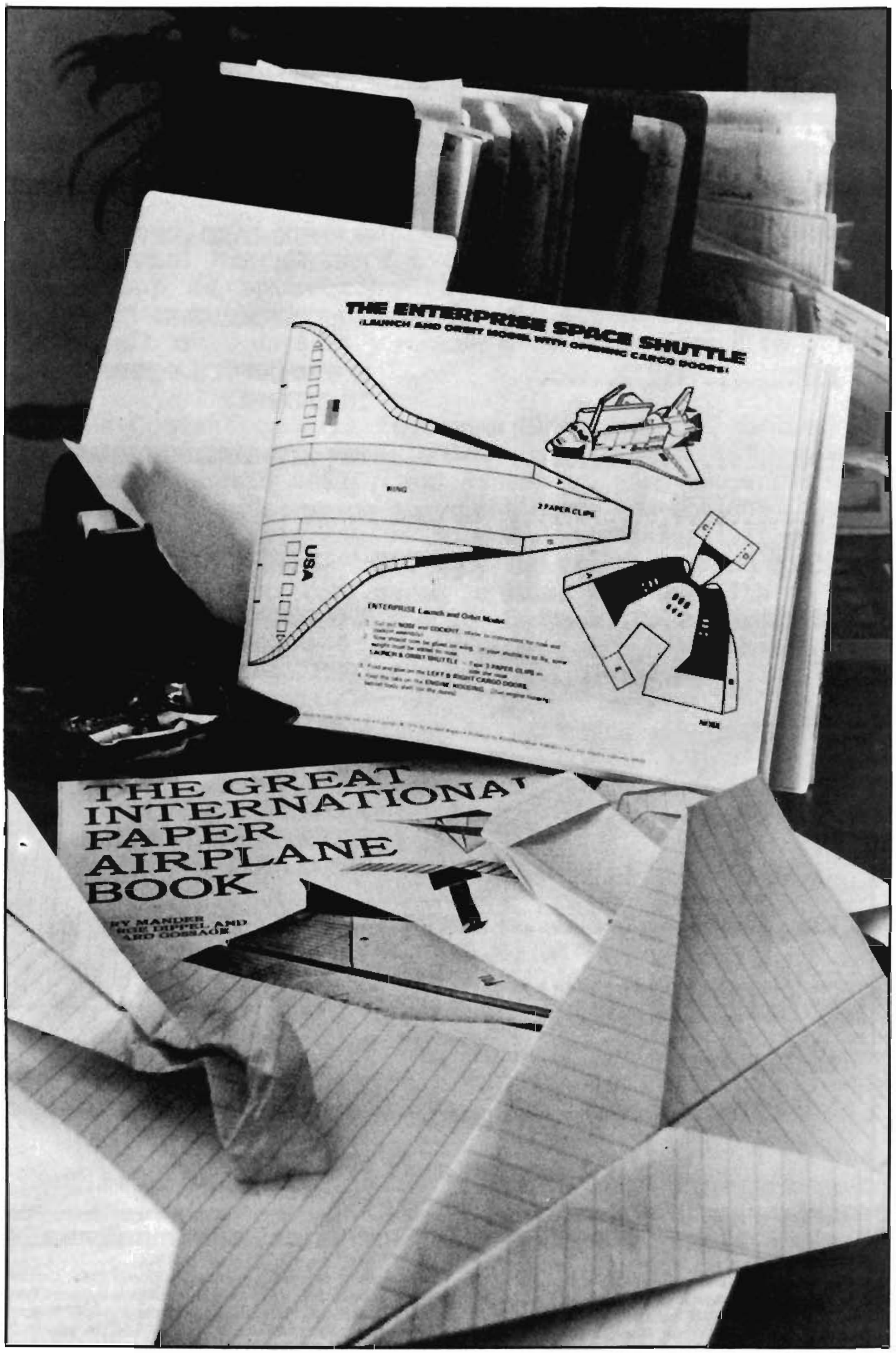
The following is a letter that was sent in by one of our users to Personal Computing Magazine. We are pleased at such a positive response. It is our policy to handle your problems as effectively as possible. Let us know if we can help you out!

Dear Editor,
As beginners in computers, we enjoy your magazine immensely. To help other beginners who often are disillusioned by lack of info, we want to pass on our enthusiastic endorsement of the COMPUCOLOR II to people. They have an excellent product at a good price and thier customer service has been unbelievable! It's been many years since we've encountered people at a company who are so helpful and understanding. They really listen and understand problems and suggestions. We received our computer BEFORE the promised date! They have a Hot Line and we've been helped time and again by both hardware and software people who understand questions the first time and give answers promptly.

W.A. Shanks
Miller Place, NY

colorcue

a publication for Compucolor users ·v. 3, · #5 ·may 1980 $1


THE ENTERPRISE SPACE SHUTTLE
(LAUNCH AND ORBIT MODEL WITH OPENING CARGO DOORS)
USA
THE GREAT INTERNATIONAL PAPER AIRPLANE BOOK

inside

- A VISIT WITH HUNTSVILLE'S COMPUNAUTS
- SCREEN SAVER
- DISSECTING A DIRECTORY
- EXECUTING FCS COMMANDS IN ASSEMBLY
- POWER LINE NOISE
- COMMUNITY ACCESS BULLETIN BOARDS
- AND MUCH MORE!

colorcue

menu

contributing to the success of this issue

USER INPUT
Tom Coughlin
Bill Greene
Tom Hudson
John Johnson
Richard Manazir
Dennis Martin
Don Miller
Edgar Swank

TECHNICAL ADVICE
Don Baines
Mike Ceigler
Knox Pannill
Fred Stevens
Rick Vick
Dave Wayman

ART DIRECTOR
Debbie Feldman

PHOTOGRAPHY
Pat Abramson (Cover)
Dennis Keim (Inside)

EDITOR
Cathy Abramson

—member international association of business communicators—

COLORCUE is published monthly for the users of the Compucolor II personal computer by Compucolor Corporation, P.O. Box 569, Norcross, Georgia 30071. Compucolor is a wholly owned subsidiary of Intelligent Systems Corp. Address all COLORCUE correspondence to our editorial office located at 100 Northcreek, Suite #250, 3715 Northside Parkway, N.W., Atlanta, Georgia 30327, (404) 261-3003. Subscriptions to COLORCUE are $12.00 per year in the U.S., Canada, and Mexico, and $24.00 elsewhere.

editor's letter

Paper airplanes are no longer just the simple fold 'em up type that I remember as a school girl -- they're a science. Our cover features **The Great International Paper Airplane Book** by J. Mander, G. Dippel, and H. Gossage published by Simon & Schuster in 1967 following the First International Paper Airplane Competition sponsored by **Scientific American**, and **The Enterprise, Columbia Space Shuttles to Color, Cut Out and Fly!** by Richard Wagner, published by Price, Stern, & Sloan, 1979.

If you are wondering about my newly acquired fascination with paper airplanes, see our cover story on the Huntsville User Group, page 6. The Huntsville Group is comprised of individuals singularly attracted to airborne things.

Announcing plans for the First Compucolor/Intecolor International User Group Convention. The Georgia Compucolor User Group is sponsoring the first combined user group convention to be held here in Atlanta, July 18 through 20. The fee for all three days is just $50. The convention site is the Perimeter North Inn.

The Convention will feature a Plant Tour and seminars on all your favorite topics: color graphics, file handling, interfacing, basic BASIC, and Assembly. And who knows what else could be in store at the First Users Convention? I have enclosed a pre-registration form in this issue. If you are interested in attending, it is very important that you fill out that form and get it back to me no later than July 7th so that I can block off enough space at the hotel to accomodate you. When I receive your pre-registration form, I will send you a complete packet of convention and hotel information.

Make plans now to meet your fellow users and join the fun! I'll see you there.

Cathy Abramson

rem

EDITOR'S NOTE

Here's an article by Tom Hudson of Springfield, MO. Tom is interested in contacting any Compucolor users in his area for information exchange.

=

SCREEN SAVER

By Tom Hudson
2369 E. Seminole
Springfield, MO. 65804
(417) 883-0727
or 493-8222

This program demonstrates a routine that allows a user to move characters around on the Compucolor screen without destroying the information already resident on the screen. It will work with any color or character, even graphics.

The routine works by coding the X and Y cursor positions into a memory address and moving the contents to FI (Foreground

Info) and BI (Background Info). FI contains the ASCII value of the foreground character. BI contains the color information for the screen position. After these values are saved, the routine is able to print the desired character at the desired location (in this case , PX and PY). The routine delays for a short time, and then restores the old screen contents by POKEing FI and BI back into screen memory.

I needed this routine for a war game I am writing. The program draws an intricate world map, and attack satellites are moved around the screen. Without this routine, the satellites would quickly destroy the map, and I'd have real trouble. By using this routine, I can move the satellites without messing up the graphic display.

One operating note: This routine will NOT work if the screen has been scrolled. To avoid this, I use an ERASE PAGE, followed by an (ESC)X (which sets up the PAGE mode, and prevents the screen from scrolling up). Normally, you will only be using this routine on non-scrolling displays, so you should have no problem.

EDITOR'S NOTE (Another solution would be to add PLOT 27,88:LOAD, name of program:RUN in the program.)

To use the routine, clear the screen and put any type of display on it (make sure the screen has not scrolled!). A good display to use for this is the High Resolution 3-D Blocks display from the graphics issue of COLORCUE (Vol. 3, #2, Feb. 1980). After the display is on the screen, load this program and RUN it. A red asterisk will move itself around the screen, and the display won't be damaged.

To adapt this technique to one of your programs, use lines 40, 70 and 100. BEFORE printing the character, move the X and Y coordinates to the PX and PY variables and execute AD = FN AD(0). Next, store FI and BI as in the routine line# 70. Now you are ready to print the character. When you are ready to restore the old screen contents, use the coding in line# 100. It's as easy as that!

By using this routine in certain games, you can make displays as complicated as you like, without worrying about ruining them.

If anyone has any questions or comments, please feel free to call or write.

LINE#	**FUNCTION**
10	This line sets up the initial values of PX and PY, the screen X and Y locations of the character to be moved around. It also sets up the XI and YI variables. These are used to increment the PX and PY values so the character will move.

20 This line sets up the address function that finds the address of the screen position indicated by PX and PY.

40 This line increments PX and PY to the next screen position.

50-60 These lines check to see if the PX and PY values have gone beyond the acceptable ranges. If so, the increments are negated.

70 This line finds the memory location for the current position of the character. Using this information, the program retrieves the information about the character currently on the screen and stores them in the FI and BI variables. BI holds the color status information. FI holds the ASCII character value.

80 Now that the old information is stored, it is O.K. to print your character. This line plots an asterisk using the blind cursor mode. Any other character could be plotted in this manner.

90 This line is a simple time-delay loop, so you can see the asterisk.

100 This line restores the old information into the screen memory. It merely POKEs the original FI and BI values back into their respective locations.

110 Now the program goes back to line 40, so it can advance the asterisk to the next position.

```
LIST

1 REM  THIS ROUTINE ALLOWS A PROGRAM TO MOVE A CHARACTER
2 REM  AROUND ON THE SCREEN WITHOUT DESTROYING THE INFO
3 REM  ALREADY ON THE SCREEN.
4 REM
10 PX= 32:PY= 16:XI= 1:YI= 1:REM SET UP X&Y VALUES, INCREMENTS
20 DEF FN AD(A)= 28672+ PX* 2+ PY* 128
30 REM ADDRESS FIND FUNCTION
40 PX= PX+ XI:PY= PY+ YI:REM INCREMENT X & Y TO NEXT POSITION
50 IF PX< 0OR PX> 63THEN XI= -XI:PX= PX+ XI
60 IF PY< 0OR PY> 31THEN YI= - YI:PY= PY+ YI
70 AD= FN AD(0):FI= PEEK (AD):BI= PEEK (AD+ 1)
71 REM GET FOREGROUND AND BACKGROUND INFORMATION
80 PLOT 3,127,PX,PY,1,42:REM PLOT AN '*' IN BLIND CURSOR MODE
90 FOR N= 1TO 50:NEXT :REM DELAY A BIT
100 POKE AD,FI:POKE AD+1,BI:REM RESTORE OLD SCREEN CONTENTS
110 GOTO 40:REM NEXT POSITION
READY
```

Steering the conversation over to things which are not airborne is impossible with this group. As we get ready to take formal shots of the Group (which somehow **never** work out!) Wubbo clowns, giving a semi-serious lecture of the, no doubt, superior abilities of his paper airplane!

From left to right, they are Stephan Bernhardt, Dick Heckman, Jim Perkins, Tony O'Neil, Wubbo Ockles, Eike Mueller, Jim Horwitz, Jim Green, Chuck Lewis, and Jeff Riggs.

A VISIT HUNTSVILLE'S

One of Compucolor's most unique groups has formed in Huntsville, Alabama, nestled between the foothills of the Appalachian mountains and the Redstone rockets.

The Huntsville User Group is entirely made up of NASA and ESA (the European Space Agency) Compucolor users, all working on the Space Lab project. It's a group with a distinct international flavor, using their machines for a wide variety of ordinary and not-so-ordinary activities. Like the rest of us, there's a wide variety of levels of expertise from just beginning to quite advanced. However, the homogeneity of their work relationships makes this group quite a contrast from most of the other groups I've seen.

The Group collected bright and early on April 26th at the home of the Group's President, Eike Mueller. First on the agenda was a demonstration of displays given by Wubbo Ockles. Dr. Ockles is a physicist by trade and a Payload Specialist by occupation. A Payload Specialist is what we used to call, in a more romantic age, an astronaut. Space Lab's Payload Specialists, 2 Americans and 3 Europeans, will be conducting a wide variety of scientific experiments while the lab is in orbit.

Wubbo and his two European counterparts are using their recently purchased CCII to simulate the experiment displays that they will be using on the mission. These displays replicate the on-board computer displays that they will actually be using. They are sending the displays back and forth to each other for training.

Eike, the NASA/ESA Liaison, demonstrates a project scheduling program which marks General Milestones

Onto serious matters, Wubbo demonstrates his Space Lab experiment simulation displays to the great interest of the Group.

for American and European activities. Tony O'Neil, head of Ground Operations, is particularly interested in this one -- just like everywhere else, apparently scheduled completion dates change frequently.

Photos
by Dennis Keim

WITH COMPUNAUTS

Eike formats this information with his Vu-Graph program, prints it out on his Paper Tiger printer, and uses the output to make transparencies for the inevitable Monday Morning Meeting. (By the way, these programs are pretty generalized and could be converted for most environments. Contact Eike if you are interested.) Eike can even plot a graph with the original projected dates, new projected dates, and final completion dates to help plan more realistic target dates in the future. On the personal side, he has developed a program to arrange and rearrange the floor plan of his home or office.

Several hours after the meeting began, the Group drifted home only to regroup at Eike's that evening for an incredible German-style barbecue. "A good time had by all" sounds hollow compared to the experience! Call or visit, it's worth the trip. Now that I have friends in high places, maybe I'll get to write a future issue of **COLORCUE** from space?

Later, Eike demonstrates the SOURCE network. The user meeting was held on the day after the aborted hostage rescue attempt, so we were all quite interested in the UPI news bulletins. That's your Editor holding up the doorway there!

MORE GROUPS!

NEW USER GROUP FORMING IN MASS./CONN. AREA

A new group is forming in the south Massachusetts -- Connecticut area. For further information contact:

Richard Manazir
13 Grandview Street
Southwick, Mass. 01077
Phone: (Day) (203) 688-1911 x4716
(Night) (413) 569-6621

CANADIAN USER GROUP

The Canadian User Group meets once every 2 months on the last Tuesday of the month at different locations near metropolitan Toronto. Fees are $15.00 per year which includes a newsletter (still un-named). There are currently about 55 members.

CANADIAN USER GROUP
Contact: Glen Davis, President
Bsmt-59 Kendal Avenue
Toronto, Ontario
Canada, M5R 1L8

EL CERRITO USER GROUP

A small but growing group according to Fred Pezok. They have amassed a sizeable list of software in their user group file, including control software for the I.D.S. 225 and BASE2 printers, for $15 each. Membership dues will be $10 plus the submission of one program after August 1st. Meetings are held on the 3rd or 4th Tuesday of the month at P.C. Computers.

THE EL CERRITO COMPUCOLOR II USERS GROUP
Contact: Fred Pezok or Frasier Hewitt
c/o P.C. Computers
10166 San Pablo Avenue
El Cerrito, California 94530
(415) 527-6657

AND TWO NEW AUSTRALIAN USER GROUPS

Both of these groups are anxious to contact other user groups, particularly in the area of program exchange, techniques, and applications.

IN THE SYDNEY AREA:
Contact: Andrew McIntosh
91 Regent Street
Chippendale, N.S.W., 2008
Australia

IN THE MELBOURNE AREA:
Contact: Neil Brandie
212 High Street
Windsor, Victoria, 3181
Australia

OTHER NEWS "DOWN UNDER"

Brian Cruse has just put out his first issue of the **Australian COLORCUE** in April. The subscription rate is $1 per issue. For further information contact:

Brian Cruse
8 Ulva Street
Bald Hills, Queensland 4036
Australia

keeping it simple

DISSECTING A DIRECTORY

By Dennis Martin
P.O. Box 4037
Omaha, Nebraska 68104
(402) 453-6826

How many of you out there have ever wondered how Compucolor builds their disk directories? Fortunately for us, the Compucolor's disk directory is set up so that it contains everything you need to know about a disk file. Let's take a closer look at how it is constructed.

As most of you know, the CCII allows you 400 (190H) blocks of disk space, with each block containing 128 bytes. When a disk is initialized, a specific portion of the disk is allocated to directory space. This is done by using the following command:

```
FCS>INI CD0:SAMPLEDISK XXXX
```

where XXXX is the number of directory blocks you want allocated in hex. Remember, this is a hex number, so if you wanted 10 blocks for the directory, you would enter 0A not 10. The CCII will allocate up to 20H blocks for the directory. If you used 20H, then you would have a maximum number of entries equal to 190! Since it would be very difficult to read a directory with that many programs on it, we usually use a much smaller number such as 06 to allow only 34 entries.

Here is a breakdown of what is on each block:

BLOCK 0

BYTE	CONTENTS
0	Current Block
1	Maximum number of directory blocks
2	Volume entry attribute (=41H)
3-12	Volume name (ASCII)
13-22	Reserved
23	File attribute
24-29	File name (ASCII)
30-32	File type (ASCII)
33	Version number in hex
34	Low order byte of starting block
35	High order byte of starting block
36	Low order byte of size
37	High order byte of size

38	Last block count
39	Low order byte of loading address
40	High order byte of loading address
41	Low order byte of starting address
42	High order byte of starting address
43	Reserved
44-64	Same as 23-43 above
65-85	Same as 23-43 above
86-106	Same as 23-43 above
107-127	Same as 23-43 above

BLOCKS 1 THROUGH XX

BYTE	CONTENTS
0	Current block
1	Maximum number of directory blocks
2-22	Same as 23-43 on Block 0
23-43	Same as 23-43 on Block 0
44-64	Same as 23-43 on Block 0
65-85	Same as 23-43 on Block 0
86-106	Same as 23-43 on Block 0
107-127	Same as 23-43 on Block 0

All entries in the Directory (except the first and last) are structured identically. The first entry contains the disk name, and the last entry contains the free space entry. That is why the formula for finding the number of entries allowed is: XX * 6 - 2.

It is very important to remember that when altering anything in the Directory, you have to alter it a block at a time. This means that if all you want to change is the 10 character disk name, you must read in the entire 128 bytes of block 0, or you can ruin a disk.

By means of a utility program, it is also possible to alter the attribute byte, program name, file type, version, and even the loading address of any file. It is also possible to create a protected file that cannot be deleted from the disk.

A final word of caution. Use extreme care whenever attempting to change anything on the disk with the WRITE command. Severe and often irreparable damage can be done to a disk through careless use of this command. For those of you that want to experiment, try using a blank disk. It could save you a lot of headaches.

NUMERIC BASE CONVERSION PROGRAM

The following program allows you to convert from decimal, hexadecimal, octal or binary into the other bases. The number to be converted is entered as a string, converted to a positive integer and then all of the results are calculated. This allows quick verification of the algorithm. The program should be self-explanatory.

```
10 REM ***** NUMERIC BASE CONVERSION PROGRAM *****
15 REM
20 REM ** WRITTEN BY:  WIZARD #6
25 REM ** TECHNICAL ADVISORS:  WIZARDS #1, #3, AND #7
30 REM
35 REM ** LAST REVISED:  MAY 18, 1980  WIZ #6
40 REM
45 REM
50 CLEAR  1000
100 CC$= "0123456789ABCDEF"
200 PLOT  6,3 : PRINT  : INPUT  "BASE (D,H,O,B) = ";NB$
205 PLOT  6,2
210 IF  NB$= "D" THEN  GOTO  2000
220 IF  NB$= "H" THEN  GOTO  3000
230 IF  NB$= "O" THEN  GOTO  4000
240 IF  NB$= "B" THEN  GOTO  5000
250 GOTO  200
260 REM
2000 REM   ** DECIMAL **
2005 NB= 10
2010 PRINT  : INPUT  "DECIMAL NUMBER = ";D$
2020 IF  D$= "" THEN  GOTO  200
2030 IF  LEN (D$)> 5 THEN  GOTO  2010
2040 SD$= D$
2050 GOSUB  6000
2060 IF  SD$= "" THEN  GOTO  2010
2070 GOSUB  7000 : REM   CALCULATE STRINGS
2080 GOSUB  8000 : REM   PRINT STRINGS
2090 GOTO  2010
2095 REM
3000 REM   ** HEXADECIMAL **
3005 NB= 16
3010 PRINT  : INPUT  "HEXADECIMAL NUMBER = ";H$
3020 IF  H$= "" THEN  GOTO  200
3030 IF  LEN (H$)> 4 THEN  GOTO  2010
3040 SD$= H$
3050 GOSUB  6000
3060 IF  SD$= "" THEN  GOTO  3010
3070 GOSUB  7000 : REM   CALCULATE STRINGS
3080 GOSUB  8000 : REM   PRINT STRINGS
3090 GOTO  3010
3095 REM
4000 REM   ** OCTAL **
4005 NB= 8
4010 PRINT  : INPUT  "OCTAL NUMBER = ";O$
4020 IF  O$= "" THEN  GOTO  200
4030 IF  LEN (O$)> 6 THEN  GOTO  4010
4040 SD$= O$
4050 GOSUB  6000
4060 IF  SD$= "" THEN  GOTO  4010
4070 GOSUB  7000 : REM   CALCULATE STRINGS
```

```
4080 GOSUB  8000 : REM   PRINT STRINGS
4090 GOTO  4010
4095 REM
5000 REM   ** BINARY **
5005 NB= 2
5010 PRINT  : INPUT  "BINARY NUMBER = ";B$
5020 IF  B$= "" THEN  GOTO  200
5030 IF  LEN (B$)> 16 THEN  GOTO  5010
5040 SD$= B$
5050 GOSUB  6000
5060 IF  SD$= "" THEN  GOTO  5010
5070 GOSUB  7000 : REM   CALCULATE STRINGS
5080 GOSUB  8000 : REM   PRINT STRINGS
5090 GOTO  5010
5095 REM
6000 REM   ** CONVERT FROM A STRING TO A POSITIVE INTEGER **
6020 SL= LEN (SD$)
6025 X= 0
6030 FOR I= 0TO SL- 1
6040 SC$= MID$ (SD$,SL- I,1)
6050 FOR  J= 1 TO  16
6060 IF  SC$= MID$ (CC$,J,1) AND  J< = NB THEN  GOTO  6100
6070 NEXT  J
6080 PRINT  : PRINT  "*** INVALID CHARACTER '";SC$;"' ***"
6083 PRINT
6085 SD$= "" : REM   FLAG ERROR
6090 RETURN
6095 REM
6100 X =  X +  ((J- 1)* (NB^ I))
6110 NEXT  I
6115 X= INT (X+ .5) : REM  CONVERT TO INTEGER
6120 RETURN
6130 REM
7000 REM   ** CALCULATE STRINGS FOR DIFFERENT BASES **
7010 BB= 10 : XX= X : GOSUB  9000
7020 DD$= NS$
7030 BB= 16 : XX= X : GOSUB  9000
7040 HH$= NS$
7050 BB= 8 : XX= X : GOSUB  9000
7060 OO$= NS$
7070 BB= 2 : XX= X : GOSUB  9000
7080 BB$= NS$
7090 RETURN
7095 REM
8000 REM   ** PRINT THE CONVERSIONS **
8010 PRINT "DEC = ";DD$,"HEX = ";HH$,"OCT = ";OO$,"BIN = ";BB$
8020 RETURN
8030 REM
9000 REM   ** CONVERT TO DESIRED BASE **
9010 NS$= ""
9020 XQ= INT (XX/ BB):XR= XX- (BB* XQ)
9030 XX= XQ
9040 NS$= MID$ (CC$,XR+ 1,1)+ NS$
9050 IF  XX< > 0 THEN  GOTO  9020
9060 RETURN
9070 REM
```

Glen Hayhurst, has sent in a hexadecimal to decimal conversion program for a programmable calculator, the HP 33E. He will be happy to provide it to you if you are interested; please send a self-addressed stamped envelope with your request.

Glen Hayhurst
9595 Pecos "14
Denver, Colorado 80221

=

advanced applications

EXECUTING FCS COMMANDS IN ASSEMBLY

It is often useful to be able to execute FCS commands from an assembly language program. This is provided for in the system software and very easy to implement. A simple example of this would be to have one program call another for a special function or task and then return to the original program.

There are three routines that are used and they are FCS, EMESS, and FCSEM. FCS is the Command Interpreter and it is used by pointing H&L to the command line to be executed and then calling FCS. The command line must end with a null byte (00). The FCS routine first searches the Command Table for the command and if it is found, the command is executed. If the command is not found or if there is some other error, the routine returns with the error code in the B register and the <Z> flag cleared; otherwise B equals 00 and the <Z> flag is set.

The EMESS routine can then be called, with or without an error. If an error is present, it will be listed to the output device using the OSTR routine (which we discussed in the September/October issue). The error conditions will be preserved through EMESS. If you wish to use FCS and EMESS regardless of an error, call FCSEM with H&L pointing to the command line.

One of the powerful features of the routine FCS is that if it is called, it will return to the calling program upon completion. This is useful if a test program is very large and can't fit into the allowed memory at one time. A main routine can be written to call particular routines as they are needed to perform desired tests. When a particular test executes to completion, control is returned to the main routine which, in turn, can call another routine located in the same area as the one just completed. NOTE: The main program must not be overwritten by a called program if the called program is to return to the main by simply executing a return (RET). The main program might reside from 8200H to 83FFH and the called routine(s) be setup to load at 8400H or higher.

Another way to use the FCS routine is to chain programs together. Suppose you designed an assembler to work with a linker. Upon completion of the assembly, the assembler would execute the linker by pointing to the 'RUN LINKER' command line and then **jump** ('JMP') to the routine FCS. The linker would execute at this point. If the linker returns when it is finished, control then passes to the caller of the assembler.

Take care to preserve the stack pointer and the stack when writing a program that will return to its caller. The program cannot be entered through the user vector, ECS ^, because the proper return address is not placed on the stack. An example would be all of the Basic Utilities (except for Fredi) which were designed to be re-entered through an Escape Vector (ECS xx) and always to return to the File Control System.

This sample program illustrates the use of the routines but is not designed to perform a specific useful task.

```
;;          ***** V6.78 SYSTEM SOFTWARE *****

FCS         EQU     25ECH    ; FCS COMMAND INTERPRETER
EMESS       EQU     262DH    ; FCS ERROR MESSAGE PROCESSOR
FCSEM       EQU     262AH    ; COMBINATION OF FCS AND EMESS


;;          ***** V8.79 SYSTEM SOFTWARE *****

FCS         EQU     0A95H
EMESS       EQU     0AD6H
FCSEM       EQU     0AD3H


            ORG     0FF00H   ; LOCATE AT TOP OF MEMORY

TEST:
            LXI     H,CMND02          ; COMMAND LINE FOR FCS
            CALL    FCS      ; RETURN TO HERE WHEN FINISHED
            CALL    EMESS    ; PRINT ERROR IF ANY (B<>0)
TEST2:
            LXI     H,CMND04
            CALL    FCS
            LXI     H,CMND06
            CALL    FCS      ; DON'T PRINT ERROR MESSAGE
            JNZ     TEST     ; IF ERROR RE-EXECUTE PROGRAM
TEST4:
            LXI     H;CMND08
            JMP     FCSEM    ; RETURN TO CALLER OF 'TEST'

CMND02:     DB      'DIR CD0:',0
CMND04:     DB      'DEV CD0:',0
CMND06:     DB      'COP SCRATC.RND TO CD1:',0
CMND08:     DB      'RUN MEMORY.PRG',0

            END     TEST
```

IBM BIT-BANGING DRIVER

Our old friend, Bill Greene, has submitted the following driver program for use on the IBM USC 1035 I/O terminal. This terminal uses the standard RS-232 outlet.

By Bill Greene
RT3 Box 00-200
Byron, Ga. 31008

```
>ASM IBMOUT
               ;       IBM BIT-BANGING DRIVER
               ;       3/24/80  BILL GREENE

               ;       START IS CALLED WITH IBM CODE IN REGISTER C

8200 (BF00)            ORG           0BF00H  ;OUT OF THE WAY IN HIGH MEMORY

BF00 79        START:  MOV           A,C     ;PXXXXXX0 TO ACC P=PARITY 0=START BIT
                                             ;XXXXXX = IBM EBCD CODE 6-LEVEL
BF01 E5        PRINT:  PUSH          H       ;SAVE SHIFT STATUS AND LINE COUNT
BF02 2609              MVI           H,9     ;BIT COUNTER-OUTPUT WILL BE 1PXXXXXX0
BF04 47        OUTBIT:MOV            B,A     ;SAVE THE BITS FOR LATER
BF05 2F                CMA                   ;NEED THE COMPLEMENT FOR IBM
BF06 E601              ANI           1       ;MASK A0 ONLY, LEAST SIGNIFICANT BIT
BF08 07                RLC                   ; X 2
BF09 C608              ADI           8       ; + 8
BF0B D304              OUT           4       ;MARK(8) OR SPACE(10) FOR RS232C
BF0D 110002            LXI           D,512   ;LOAD LOOP COUNTER
BF10 CD1CBF            CALL          LOOP    ;FOR 7.43MS DELAY
BF13 78                MOV           A,B     ;GET REMAINING BITS
BF14 37                STC                   ;SET CARRY FOR STOP BIT(1)
BF15 1F                RAR                   ;1PXXXXXX, 11PXXXXX, 111PXXXX, ETC.
BF16 25                DCR           H       ;COUNT THE BIT
BF17 C204BF            JNZ           OUTBIT  ;GET ANOTHER ONE
BF1A E1                POP           H       ;RESTORE STATUS
BF1B C9                RET                   ;GO HOME

BF1C 1B        LOOP:   DCX           D       ;DECREMENT D,E REGISTER
BF1D 7A                MOV           A,D     ;LOOK AT D
BF1E FEFF              CPI           255     ;LOOKING FOR 65535, ESSENTIALLY -1
BF20 C21CBF            JNZ           LOOP    ;WASTE MORE TIME
BF23 C9                RET

BF24 7534B4F5 TABLE: DB              75H,34H,0B4H,0F5H  ;IBM CODE
BF28 68C35864          DB            68H,0C3H,58H,64H   ;FROM ASCII 33 TO ASCII 96
BF2C 540443F6          DB            54H,04H,43H,0F6H
BF30 81B7E2D4          DB            81H,0B7H,0E2H,0D4H
BF34 A090F088          DB            0A0H,90H,0F0H,88H
BF38 E8D8B884          DB            0E8H,0D8H,0B8H,84H
BF3C E4087010          DB            0E4H,08H,70H,10H
BF40 20386282          DB            20H,38H,62H,82H
BF44 2313730B          DB            23H,13H,73H,0BH
BF48 6B5B3B07          DB            6BH,5BH,3BH,07H
BF4C 67615131          DB            67H,61H,51H,31H
BF50 49291979          DB            49H,29H,19H,79H
```

```
BF54 45255232      DB          45H,25H,52H,32H
BF58 4A2A1A7A      DB          4AH,2AH,1AH,7AH
BF5C 462676E2      DB          46H,26H,76H,0E2H
BF60 76370102      DB          76H,37H,01H,02H

BF64 803C   STATUS:DB          128,60;SHIFT FLAG AND LINE COUNT

BF66 (BF00)        END         START

SYMBOL TABLE

LOOP....BF1C OUTBIT..BF04 PRINT...BF01 START...BF00 STATUS..BF64
TABLE...BF24
```

=

nuts & bolts

power,

Electrical noise is a major problem in microcomputers. Generated within a computer itself, it can in many cases interfere with nearby radio and TV reception and even with the computer's own operation. Many unexplained microcomputer malfunctions can, in fact, be traced to the external noise. All microcomputers are more prone to electrical noise problems than large mainframe computers. Let us explore one source of noise and its effect on microcomputer operations and suggest methods of reducing or eliminating the problem.

Nearby lightning strikes generate spikes of the highest amplitude, but strikes a mile or more away can also cause trouble. This noise appears on an oscilloscope as a rapid-fire series of narrow, high voltage spikes. It can enter the power system either from a direct strike on power distribution equipment or by induction. Since power lines may be strung overhead for long distances, they efficiently pick up electrostatic and electromagnetic fields radiated by lightning strikes and conduct the resulting spikes into your home.

Probably the power line noise with the next highest amplitude is created by switching off an inductive appliance, allowing energy stored in the magnetic field of an inductive device to release suddenly as the field collapses. For a few dozen microseconds, a very high voltage is built up and may arc across the open switch contacts, feeding a high voltage pulse with a fast rise time into the power line. AC induction motors and transformers -- even those in fluorescent lamp fixtures -- are the most highly inductive. Large motors, such as those used in refrigerators and air conditioners, create an electrical noise even when they switch on because an internal starting winding switches off when they reach operating speed. Industrial plants with hundreds of very large motors and transformers constantly switching on and off present particularly hostile electrical environments for computers.

While these sources are responsible for occasional high amplitude spikes on the power line, universal motors (those with brushes) and lamp dimmers generate lower amplitude, but more continuous noise. The obvious source of noise in universal motors originates with armature brushes that constantly spark during motor operation. Lamp dimmers with silicon-controlled rectifiers (SCRS) or triacs generate a small amount of noise on every alternation of the AC line because of the rapid turn-on characteristics of these semiconductors.

One might reasonably ask how power line noise can get into a computer's logic circuitry if its very narrow pulse widths cannot pass through the power transformer and rectifier diodes to actually change the DC supply voltages (large power-supply filter capacitors prevent this). Noise is capacitively coupled from the power supply's primary wiring directly into the regulated DC output wiring and then into the logic. In some computers with the primary wiring routed close to the logic circuitry, noise bypasses the power supply altogether and couples directly into the logic. It is not difficult to visualize how even a fraction of a picofarad of coupling capacitance can transfer more than 1 volt (the typical noise-immunity specification for most TTL and MOS logic) of the original 1000-volt spike into the logic circuitry and possibly cause an error. NOTE: Most computers are sensitive to noise only during part of their BUS and memory cycles; hence, it is possible for the computer to ignore the majority of noise spikes.

line

Preventing power line noise from entering a computer is relatively easy -- simply use a noise filter between the AC outlet and the computer's line cord. Filters are most effective if located inside the computer's cabinet, right at the point where the power cord enters. Even if installed at the plug end of the computer's line cord, however, they can be very helpful in excluding power line noise.

While filters are available from a number of sources, you might want to make your own filter circuit. A good power line filter consists basically of two L-section low-pass R-F filters (to provide both common and transverse mode noise rejection), with inductors designed to handle several amperes of continuous current. Metal-oxide varistors (MOVS) are included to prevent possible damage from direct local lightning strikes but do not otherwise contribute to filtering out noise. (MOVS act like high-power zener diodes, shorting out electrical spikes that exceed about 300-volts in amplitude.)

noise

A properly grounded three-wire AC outlet and computer line cord are required for any noise filter to function properly. With non-grounded outlets, filtering efficiency will be very low and current leakage through the filter's capacitors could create a mild shock hazard.

AC-line filters cannot cope with problems caused by low-frequency power surges and dips long enough in duration to affect the computer's DC voltages. The only defense against this type of noise is a wide line-voltage range computer power supply or an expensive external AC voltage regulator.

product showcase

THE COMPUCOLOR MONITOR

The Compucolor Monitor disk is used to display and alter the memory contents of the Compucolor II. Commands are available providing access to memory in several ways and the command syntax is really quite simple. A command is one character followed by one or more parameters separated by spaces. There are no spaces immediately after the command character itself. The form looks like this:

.Cxxxx yyyy zz

C is the command, xxxx is the first parameter, yyyy is the second parameter, and zz is the third (the maximum number of parameters allowed). All parameters are in hexadecimal format.

List of the available commands and their formats:

A -- Display memory contents in ASCII.	.Assss eeee
D -- Display memory contents in hex.	.Dssss eeee
E -- Exit program, return to CRT mode.	.E
F -- Fill memory with a constant.	.Fssss eeee cc
G -- Go to memory location with optional breakpoints.	.Gssss bbbb
M -- Move memory.	.Mssss eeee dddd
S -- Substitute memory contents.	.Sssss

There are three programs included on the MONITOR disk, two LDA-type files called MONT.LOW and MONT.HGH. The two MONT files are assembled at 0H and 100H. The disk also includes a MENU program which creates a MONT.PRG program at any RAM memory location you desire. MENU compares the two files and can be used to relocate any assembly language program having identical LDA versions assembled at 0H (LOW) and 100H (HGH). It does not have a DS as its last statement.

The MONITOR disk is now available from your Compucolor dealer or from the factory for $19.95.

EDITOR'S NOTE

As a service to our readers, Compucolor Corporation publishes information sent to us by other manufacturers of products, techniques, or scientific and technological developments which can be used in conjunction with the Compucolor computer. However, in many cases, Compucolor Corporation has not had the opportunity to examine or test these products thoroughly. Therefore, we neither endorse nor assume any responsibility of liability for the proper functioning of products not directly manufacturered by the Compucolor Corporation. We hope you will find these products of interest, but we encourage you to investigate carefully before you buy.

COMPUWORLD ANNOUNCES ACCOUNTS RECEIVABLE & MAILING LABEL PACKAGES

Compuworld, Inc. announces the release of two business packages for use on the Compucolor. They are Accounts Receivable and Mailing List. Accounts Receivable will support 100 accounts and 500 invoices per disk side. AR will provide statements for all accounts with open invoices; invoice aging and account status; total monies owed for all accounts; and mailing labels by two parameters. Accounts Receivable is available for $150.

The Mailing List package allows entry, correction, and deletion capabilities. You can print your labels according to a wide variety of parameters. The Mailing List is available for $75.

Both packages run on a 32K Compucolor and do not require special keyboards, lower case or add-on drives. To order in New York State, add 7% Sales Tax. Dealer inquiries welcome. For further information contact:

Compuworld, Inc.
125 White Spruce Boulevard
Rochester, N.Y. 14623
(716) 473-5790

MICRO DATA BASE SYSTEMS RELEASES SOFTWARE FOR THE COMPUCOLOR

Jerry Kidd from Micro Data Base Systems, Inc. has written saying:

"Available now for the Compucolor 16K+ is F.R. Ruckdeschel's Instrument Flight Training Program, Tax Program (1040A only), Dictionary Listing with String Search Programs (a particularly useful tool for those who write educational programs), IQ Builder Kit, Basic Grammar Kit, Basic Algebra Kit (the kits are continually being expanded), Word Scramble Drills and some interesting new games; Golf (play some of the best courses in America), Space Pilot (fly assorted spacecraft in combat), Stockmarket, Super Hammurabi, and other programs."

A free brochure is available from:

Micro Data Base Systems, Inc.
P.O. Box 248
Lafayette, Indiana 47902

input

NO CHEATERS!

John Johnson
561 W. Enid Avenue
Mesa, Arizona 85202

After playing "Acey Ducey" on the Chess diskette for awhile and being a naturally curious person, I tried inputting a seemingly illegal value for a bet with some interesting results. It appears that the program will accept a negative bet and if the bet loses, the negative value is subtracted from your score, in effect, adding to your total. If you are like me, when I get frustrated after losing a lot and I know of a way to "fudge" a bit, I will use it. To resist this temptation, I modified the program by adding the following lines:

```
666  IF M>=0 THEN 670
667  PLOT 3,0,30,11,3,0,30:PRINT "HA,HA!  YOU CAN'T DO THAT
     ANYMORE."
668  FOR I=1 TO 1000:NEXT:PLOT 3,0,30,11:GOTO 650
```

This will not allow values less than 0 and will remind you that "cheaters never prosper", well not anymore anyway!

Tom Coughlin
c/o Elliott, Powell,
Baden & Baker, Inc.
1521 S.W. Salmon Street
Portland, Oregon 97205

Tom Coughlin is working in the life insurance business and is specifically interested in locating general business software and software developed for use in the insurance business. Please contact him if you have any information.

THE COPY COMMAND

Richard Manazir
13 Grandview St.
Southwick, Mass 01077

Instructions for the complete use of the COP command for CCII, V6.78 software. The Copy Command does not allow the user to copy from the default drive to the other drive. This situation is true regardless of which drive is the default drive.

When drive 0 is the default drive, files can be copied from drive 1 to drive 0. To copy files from drive 0 to drive 1, do the following:

```
ESC D (Enter FCS)
FCS> DEV 1:
DEFAULT DEVICE= CD1:
FCS> COP 0:filename.type(;version) TO 1:
```

Use the same process when the default drive is drive 1.

EDITOR'S NOTE

(If the destination device is not stated, then the destination device is the default device.)

This method is quite simple and eliminates the need for using external programs such as COPY on the Sampler disk.

Also programs can be loaded from drive 1 when drive 0 is the default drive (this pertains to programs written in BASIC). Simply type:

```
LOAD "1:filename" or
LOAD "0:filename"
```

The same holds true for the SAVE command.

=

FURTHER HINTS ON PRINTER PROBLEMS

Edgar W. Swank
5515 Spinnaker Drive, #4
San Jose, California 95123

This article adds more information to the ATTN/BREAK article in the February **COLORCUE**, which told how to avoid printing extraneous lower-case "y"s when using a printer with a hardware handshaking modification installed on the Compucolor II by using OUT 8,4 to disable the keyboard. Here are some additional hints:

1. Despite the previous article, it IS possible to enable the keyboard to accept an ATTN/BREAK occasionally without affecting printed output.

You now get to use the BASIC command you haven't used yet, I'll bet. Before you enable the keyboard, you must wait for the printer handshake line to quiesce. You can do this by issuing in BASIC:

```
WAIT 1,128,128:OUT 8,255
```

Of course you must issue OUT 8,4 before sending any more data to the printer.

2. OUT 8,4 will NOT WORK if you issue any BASIC FILE commands FILE, GET, PUT, or LOAD/SAVE) while the printer is active. That is because these commands, for some reason, override your OUT 8,4 and re-enable the keyboard. Fortunately, you can overcome this problem by issuing WAIT 1,128,128 BEFORE any file commands, and OUT 8,4 again before sending any more printer data.

3. Be sure to press CPU RESET before entering ESC G when you want to use DIR to print a directory. This ensures that keyboard input is directed to the screen, while FCS output goes to the printer. The "y"s from the handshake line will then appear on the screen, rather than on the printer. If you fail to press RESET, sometimes ESC G will cause both keyboard input and FCS output to be sent to the printer. NOTE: After RESET you may have to reset the BAUD RATE unless your printer can be set to 9600.

4. When you want to LIST a BASIC program, try the following procedure, which avoids the necessity of holding down two keys while the printer is active:

a. RESET the machine and LOAD the program to be listed. Set Baud Rate, if necessary.

b. Enter POKE 33289,<the line length of your printer>

c. Enter the following on ONE line:

```
    PLOT 27,13:OUT 8,4:LIST:WAIT 1,128,128:PLOT 10,12:WAIT
1,128,128:POKE 33265,0
```

The PLOT 10,12 and the extra WAIT seem to be necessary to ensure that the last line is printed.

attn/break

COMMUNITY ACCESS BULLETIN BOARDS

Sent By
Don Miller
112 Marble Dr.
Rochester, N.Y. 14615
(716) 663-1175

COMPLIMENTS OF PEOPLES' MESSAGE SYSTEM, SANTEE CA.
LAST UPDATED 03/07/80
*24 MEANS 24-HOUR SERVICE

ABBS A.C.E.S., FT. LAUDERDALE FLA..............	(305)	566 0805
ABBS AGS, AUGUSTA GEORGIA......................	(404)	733 3461
ABBS AKRON DIGITAL GROUP, AKRON OHIO *24.......	(216)	745 7855
@BBS AMRAD, MCLEAN VA..........................	(703)	893 9474
ABBS APPLE CRATE I, VANCOUVER WASHINGTON.......	(206)	524 0203
@BBS APPLE CRATE II, VANCOUVER WASHINTON.......	(206)	244 5438
ABBS APPLE-MED, IOWA CITY IOWA.................	(319)	353 6528
ABBS APPLE BIN, TACOMA WASHINTON...............	(206)	937 0444
A@BS BAILEYS COMPUTER STORE, AUGUSTA GEORGIA...	(404)	790 8614
@BBS BOSTON, BOSTON MASSACHUSETTS..............	(617)	354 4682
@BBS CCNJ, POMPTON PLANS NEW JERSEY............	(201)	835 7228
A@BS CHICAGO, CHICAGO ILLINOIS.................	(312)	622 9609
@BBC COMPUTERLAND, ARLINGTON HEIGHTS ILLINOIS..	(312)	255 6489
ABBS COMPUTER CENTER, BIRMINGHAM ALABAMA.......	(205)	945 1489
ABBS COMPUTER FORUM, SANTE FE SPRINGS..........	(714)	739 0711

ABBC COMPUTER LAB, MEMPHIS TENNESSEE	(901) 761 4743
ABBS COMPUTER MERCHANT, SAN DIEGO CA	(714) 582 9557
ABBS COMPUTER STORE, SANTA MONICA CA	(213) 394 1505
ABBS COMPUTER STORE, CINCINNATI OHIO	(513) 874 2283
@BBS COMPUTER WORLD, IRVINE CA *24	(714) 751 1422
ABBS COMPUTER WORLD, WESTMINISTER CA	(714) 898 1984
ABBS CYBERIAN SYSTEMS, ATLANTA GEORGIA	(404) 939 5429
ABBS ELTREX, LOS ANGELES CA *24	(213) 826 0325
@BBS DESERT TECHNOLOGY, PHOENIX ARIZ *24	(602) 957 4428
@BBS DETROIT, DETROIT MICHIGAN	(313) 477 4471
ABBS DOWNERS GROVE, DOWNERS GROVE ILLINOIS	(312) 964 7768
ABBS DUBUQUE, DUBUQUE IOWA	(319) 557 9618
ABBS FORT WALTON BEACH, DESTIN FLA	(904) 243 1257
BBS FRONTIER COMPUTING, LOGAN UTAH	(801) 753 6800
ABBS GLTC.DV1, HOLLYWOOD FLA	(305) 989 9647
ABBS HAWTHORNE, HAWTHORNE CA *24	(213) 675 8803
@BBS HOUSTON, HOUSTON TEXAS	(713) 977 7019
ABBS F, ANAHEIM CA	(714) 772 8868
ABBS LLINI MICROCOMPUTER, NAPERVILLE, ILL.	(312) 420 7995
A@BS A COMPUTER STORE, DALLAS TEXAS	(214) 634 2668
BBS ORSMEYER ELECT., HUNTINGTON BEACH CA *24	(714) 962 7979
ABBS HOUISVILLE, LOUISVILLE KENTUCKY	(502) 245 8288
ABBS MC COMPUTER STORE, SANTA MONICA CA	(213) 828 3400
ABBS MIAMI, MIAMI FLA	(305) 821 7401
A@BS MICHIGAN APPLE CLUB, DETROIT MICHIGAN	(313) 569 2063
@BBS IINNEAPOLIS, MINNEAPOLIS MINNESOTA *24	(612) 929 8966
ABBS NJAUC, MARLTON NEW JERSEY	(609) 983 5970
ABBS OKS, OKLAHOMA CITY OKLAHOMA	(405) 528 8009
@BBS ORIGINAL APPLECORPS, LOS ANGELES CA *24	(213) 340 0135
ABBS PALM BEACH, WEST PALM BEACH FLA	(305) 689 3234
@BBS PALO ALTO, PALO ALTO CA	(415) 493 7691
ABBS PASADENA, PASADENA CA *24	(213) 799 1632
ABBS PCNET, LOS ALTOS CA	(415) 948 1474
@BBS PERIPHERALS UNLTD. INC., SIGNAL HILL CA	(213) 424 3506
ABBS PERSONAL COMP. OD CHICAGO, CHICAGO ILL.	(312) 337 6631
@BBS PHOENIX, PHOENIX ARIZONA	(602) 886 0258
ABBS PROGRAMMA INTL., BRENTWOOD CA *24	(213) 395 1592
@BBS RAINBOW COMPUTING, LOS ANGELES CA	(213) 349 5728
A@BS JAN ANTONIO, SAN ANTONIO TEXAS *24	(512) 657 0779
@BBS SAN FRANCISCO, SAN FRANCISCO CA	(415) 661 0705
ABBS SANTA MONICA, SANTA MONICA CA	(213) 396 3905
ABBS STATEN ISLAND, NEW YORK NEW YORK *24	(212) 448 6576
ABBS TORRANCE, TORRANCE CA	(213) 316 5706
@BBS 27-MICRO, LOS ANGELES CA	(213) 276 4276
ABBS VAN NUYS, VAN NUYS CA	(213) 787 4004
ABBS WEST WASHINGTON, ELMA WASHINGTON	(206) 482 5134
@BBS WYCKOFF, WYCKOFF NEW JERSEY	(201) 891 7441
ABBS YOUNGS ELECT SVC., COLLEGE STATION TEXAS	(713) 693 8080
AVIONICS 80, KANSAS CITY MISSOURI	(913) 782 5115
BILLBOARD 80, SAN JOSE CA	(408) 263 0248
CBBS ATLANTA, ATLANTA, GEORGIA *24	(404) 394 4220
CBBS BOSTON, BOSTON MASSACHUSETTS *24	(617) 963 8310
CBBS CAMBRIDGE, CAMBRIDGE, MASSACHUSETTS *24	(617) 864 3819
CBBS CHICAGO, CHICAGO ILLINOIS *24	(312) 528 7141
BBBS COLUMBUS, COLUMBUS OHIO *24	(614) 272 2759
CBBS DETROIT, DETROIT MICHIGAN *24	(313) 288 0335
CBBC L.I. COMP ASSOC, LONG ISLAND NEW YORK	(516) 938 9043

BB NASHVILLE, NASHVILLE TENNESSEE *24	(615)	254 9193
CBBS NEW YORK, ENDICOTT NEW YORK *24	(607)	754 5571
CBBS NW, PORTLAND OREGON *24	(503)	646 5510
BBS SAN DIEGO, SAN DIEGO CA	(714)	571 5550
CBB SAN FERNANDO, LOS ANGELES CA	(213)	843 5390
BBBS SANTA CLARA, SANTA CLARA CA	(408)	241 1956
CBBS TERRY FOX, WASHINGTON D.C. *24	(703)	734 1387
@BBS VANCOUVER, VANCOUVER BRITISH COLUMBIA *24	(604)	687 2640
CBBS WASHINGTON, WASHINGTON D.C. *24	(703)	281 2125
MMS CHICAGO, CHICAGO ILLIONIS	(312)	767 0202
COMM-80 ORANGE COUNTY, FULLERTON CA	(714)	526 3687
COMPUSYSTEMS, COLUMBIA SOUTH CAROLINA	(803)	771 0922
ENGINEER-80, OLATHE KANSAS	(913)	764 1520
FAMILY HISTORIANS FORUM 80, FAIRFAX VA *24	(703)	978 7561
FORUM-80 AUGUSTA, AUGUSTA GEORGIA	(803)	279 5392
FORUM-80 DALLAS, DALLAS TEXAS	(214)	288 4859
FORUM-80 DUNSTABLE, DUNSTABLE MASS. *24	(617)	649 7097
FORUM-80 FT. LAUDERDALE, FT. LAUDERDALE FLA	(305)	772 4444
FORUM-80 FT. WORTH, FT. WORTH TEXAS *24	(817)	923 0009
FORUM-80 KANSAS CITY, KANSAS CITY MISSOURI *24	(816)	861 7040
FORUM-80 LAS VEGAS, LAS VEGAS NEVADA	(702)	873 9491
FORUM-80 LOS ANGELES, LOS ANGELES CA	(213)	566 8035
FORUM-80 MEMPHIS, MEMPHIS TENNESSEE *24	(901)	276 8196
FORUM-80 SAN FRANCISCO, SAN FRANCISCO CA *24	(415)	348 2139
FORUM-80 ST. LOUIS, ST. LOUIS MISSOURI	(314)	838 7784
FORUM-80 TAMPA, TAMPA FLA	(813)	223 7688
FORUM-80 VENTURA COUNTY, CAMARILLO CA *24	(805)	484 9904
FORUM-80 WASHINGTON D.C., WASHINGTON D.C	(202)	337 4694
FORUM-80 WELLESLEY, BOSTON MASSACUHSSETTS	(617)	431 1699
FORUM-80 WICHITA, WICHITA KANSAS	(316)	746 2078
FORUM-80 WICHITA FALLS, WICHITA FALLS TEXAS	(817)	855 3916
MARKET 80, KANSAS CITY MISSOURI	(816)	931 3135
MEDICAL FORUM-80, MT. CLEMENS MICHIGAN	(313)	465 9531
NEW ENGLAND COMP. SOC., MAYNARD MASS	(617)	897 0346
PEOPLES' MESSAGE SYSTEM, SANTEE CA *24	(714)	449 5689
PET BULLETIN BOARD, YPSILANTI MICHIGAN *24	(313)	484 0732
POTOMAC MICRO MAGIC INC., FALLS CHURCH VA	(703)	750 0930
REMOTE NORTHSTAR, ATLANTA GEORGIA *24	(404)	939 1520
REMOTE CP/M MINICBBS, ROYAL OAK MICHIGAN *24	(313)	588 7054

=

SPECIAL INVENTORY SURPLUS SALE

We had such a good response to our inventory sale in last month's issue of **COLORCUE** that we decided to offer these parts as well. These bargain prices can't be beat. You can take advantage of this sale by just filling in the quantity and making out a check or money order to Compucolor for the total amount. In Georgia, please add 4% sales tax.

ITEM	COST PER ITEM	QUANTITY
(#300010) .02 UF 1600V	.05	
1N4731	.10	
2N5771 PNP	.10	
(#110097) IC HEAT SINK	.10	
(#300055) 2,500 UF 15V	1.00	
3.3 ohm 2W	.05	
150 ohm 1/4W	.03	
4.3K 1/4W	.03	
NUTS PIX TUBE MOUNT	10/.10	
SPACE BAR TABS	.10	
LOGO CLIPS	.03	
FOCUS RESISTOR	.35	
TUBE SOCKET CLIPS	.07	
(#900033) YOKE CLAMP SCREW	10/.10	
4-40 X 7/8 X 7/8 BRASS HEX SPACER	.10	
(#200022) 680 ohm 1/4W RESISTOR	.03	
220 UF 16V	.15	
(#120067) 8-32 X 3/8 FLAT HEAD	10/.10	
.068 UF 250V	.10	
007-9817178 CONNECTOR	1.10	
(#200043) 39K 1/4W	.03	
(#600083) 6 PIN CONNECTOR	.10	
4 PIN CONNECTOR	.10	
10 UHY CHOKE	.10	
(#600013) 3 PIN TRANSISTOR SOCKET	.15	
(#500120) CRT CHIP 5027-000 UF9	2.00	
C1702A	2.00	
(#120049) POST NUT	.02	
(#120032) 10-32 X 1/2 FLAT HD SCREW	10/.10	
(#120041) 8-32 HEX NUT	.05	
(#110010) FUSE CLIP	.02	
(#120022) 4-40 X 3/4 TRUSS HD	10/.10	
4-40 X 3/4 PAN HD	10/.10	
8 X 3/4 HEX HEAD	10/.10	
(#110048) SPRING FUSE CLIP	2/.10	
(#500055) 81LS96	1.00	
ANSLEY 609-255	1.00	
(#120075) 6 X 3/16 SPACER	2/.10	
(#120059) 6-32 X 7/8 SCREW	10/.10	
(#120066) #10 WASHER FLAT	10/.10	
(#120062) 4-40 X 1/8 ROUND HEAD SCREW	10/.10	
(#200207) 1K 2W RESISTOR	.05	
EXPERIMENTAL CASE	5.00	
BELL FOR COMPUCOLOR	14.95	
MYSTERY PACKAGE (nuts, bolts, screws, transistors, capacitors, ICs, diodes, etc.)	5.99	
SURPLUS 8000 INTECOLOR COMPUTERS (without cards)	$200 - $600	

We still have a good quantity of the items listed on last month's inventory list.

	COST PER ITEM	QUANTITY
Assorted key caps and space bars (list the keys you need to the left of the quantity column.)	$ .10	
Compucolor disk drive face plates	$ .50	
1K 2W 10% resistors	$ .05	
38,000 MFD 15 VDC capacitors	$3.66	
610 MFD 15 VDC capacitors	$1.99	
1100 MFD 100 VDC capacitors	$1.48	
8700 MFD 40 VDC capacitors	$1.46	
3500 MFD 30 VDC capacitors	$1.46	
1702 EPROMS	$2.00	
Of special interest -- motor speed controls and video blanking circuits (either separate or combined on one board). The motor speed control circuit locks disk speed at 300 rpm. Brand new, no warranty. Will not work with the formatter. Motor Speed Control without video blanking only	$30.00	
with video blanking	$35.00	
375 RCA 8-track stereo recorder/players (without case) for sale. Look into this for mass storage, nothing beats it for the price. A really great buy at only	$5.00 each!	
ADI Keyboards -- we will refund purchase price to the 1st person who interfaces this to the CCII! It can be done! The ADI has a detachable keyboard which presents the standard ASCII 4-level code. The keyboard keys are opticaly encoded by means of photo transistors, a light source, and shutters attached to the keys. There are no switches to wear out and the unit is RFI free. The keyboard does not provide 2-key rollover.	$25.00	

These parts may be slightly used or be brand new surplus inventory -- they are guaranteed to work, and all parts are priced very low!

YOUR NAME: ____________________

ADDRESS: ____________________

____

____

MAIL TO:

Compucolor Corporation
225 Technology Park/Atlanta
Norcross, Georgia 30092
Attn: Reclamation Department

Effective 6/1/80 TERMS: CASH WITH ORDER

HOW TO SUBSCRIBE AND ORDER BACK ISSUES

New owners are entitled to a six month's free subscription to **COLORCUE** as soon as their warranty card has been returned. Following the first six month's free subscription, the subscription rate is $12 per year in the U.S., Canada, and Mexico and $24 throughout the rest of the world.

Each quarter, back issues are combined into one issue covering that period. Quarterly issues may be ordered for $5 each. Once the quarterly issue is ready, individual issues will not be sold separately. During the current quarter, individual issues are available for $2 each.

Below is a list of all back issues of **COLORCUE.** To order, fill out the attached order blank, enclose your check or money order (no cash, please), and mail your order to:

Cathy Abramson
COLORCUE
100 Northcreek, Suite #250
3715 Northside Parkway, N.W.
Atlanta, Georgia 30327

****

Please send me the issues I have checked below:

___ Volume 1, Nos. 1, 2, 3 October-December 1978

___ Volume 2, Nos. 1, 2, 3 January-March 1979

___ Volume 2, Nos. 4, 5 April-May/June 1979

___ Volume 2, Nos. 6, 7, 8 August-November 1979

___ Volume 3, Nos. 1, 2, 3 Dec./Jan. - March 1980

___ Volume 3, No. 4 April 1980

Please enter my subscription for ___ year(s) beginning ________.

I have enclosed my check or money order in the amount of $___. Thank you.

NAME:____

ADDRESS:_____________________________________

CITY:_______________ STATE:_____________ ZIP:____________

Compucolor® Corporation
Intecolor Drive
225 Technology Park/Atlanta
Norcross, Georgia 30092
Telephone 404/449-5996