ENGINEERING NOTES
on
Radio Shack Color Computers

September 1986
Vol. 3 No. 8

$1.95

DYNAMIC COLOR NEWS

PROGRAMS

* DISK FILE UTILITY
* ANTENNA DESIGH
* ML DATA MOVE
* ML ASCII OUTPUT

INSTRUCTIONAL SERIES

* HAM RADIO & COMPUTERS (Part 2)
* ML PROGRAMMING (Part 5)
* WRITING PROGRAMS (Part 18)
* INTERFACING COMPUTERS (Part 8)

SERVICES

* NEW PRODUCTS
* PRODUCT REVIEWS
* QUESTIONS & ANSWERS
* OERATING HINTS

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

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

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

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

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

DYNAMIC COLOR NEWS

September 1986

Editor and Publisher
Bill Chapple W4GQC

Secretary
Dean Chapple

CONTENTS

256K & 512K MEMORY UPGRADES

If you have a 64K computer with sockets for the SAM and 4164 chips then you can update it to 256K or 512K. The ramdisk allows programs to be retain within your computer and loaded as needed. Features include:

* 40 Track Single Disk Swap Can serves as second drive.
* Fast 35/40 Track Ramdisk (2 Ramdisks with 512K).
* 32K to 200K printer spooler (400K with 512K RAM).
* More then 30 PMODE 4 screens at once.
* Pager configures computer for 8 (16 with 512K) 32K pages.
* OS-9 Ram Disk 35-40 track single sided or 40 track double sided with 512K.
* Memory is protected when the computer is reset.
* Solderless installation.
* Miniature toggle switch can force computer into 64K mode.
* Compatible with all software.

Software is supplied on tape or disk execept OS-9 is not available on tape. Specify your choice when ordering. Assemblies are complete ready to install with memories and 64K mode switch. Order ME-16 for 256K assembly, ME-14B provides extra 256K for ME-16. ME-16A for 512K assembly.

ME-16 - 256K RAM	$99.95
ME-14B - Second 256K for ME-16	79.95
ME-16A - 512K RAM	169.95

128K UPGRADES

ME-10A Upgrades 64K Korean Computers to 128K. $49.95

ME-12 - Upgrades all 64K computers with 4164 memory chips to 128K. $49.95

VIDEO REVERSER

An integrated circuit that mounts on the 6847 and reverses the video reducing eye strain. Minor soldering for CC-2. $9.95

MEMORY SAVER

(Uninterrupted Power Source)

Our UPS saves your programs from being lost due to power failures by providing power to the memories from its battery. The assembly consists of a control circuit, battery, miniature toggle switch and a light emitting diode (LED). The control circuit and battery mount under the keyboard or can be mounted outside. The switch enables the UPS and the LED glows when power is available. For all computers with 5 volt memories. $59.95

MEMORY MANAGER

(New Product)

A complete set of software for managing the second 32K memory bank for 64K and larger computers. Run Basic programs in both banks, continue a basic program from one bank to the other, use the second bank for a RAM DISK, configure the computer for the all RAM mode and store programs in the upper memory. $27.95 cassette, $29.95 disk.

24 hr phone. Checks, VISA & MC cards. Add $3 ship.

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

ML PROGRAMMING
(Part 5)

In this series we are showing how to write machine language programs and subroutines. Machine language subroutines can be linked to basic with the USR or EXEC commands. The advantages of machine language programs are speed and compactness. Machine language programs make the computer run at its fastest rate. Basic commands have to be converted to machine language instructions which makes it slower. However basic programs are much easier to write and debug than machine language programs. Therefore basic can be used for the program structure and machine language subroutines called when speed is needed.

To write machine language programs an assembler is required. An assembler allows instructions to be written in mnenomics. After the program is written it is converted or assembled into machine language codes. A disassembler displays the mnenomics for machine language instructions.

ADDRESSING MODES

Perhaps the hardest thing about learning machine language programming is understanding addressing modes. Let's take another look at using addressing modes and the number of bytes required for each mode.

First of all let's look at the immediate mode. If you want to load a register with a value then use the immediate mode. The 8 bit registers require one byte for the value and the 16 bit registers require 2 bytes. The format is as follows:

Memory	Function
M	Immediate instruction (8 bit)
M+1	Value for 8 bit register
M+2	Immediate instruction (16 bit)
M+3	MS for 16 bit register
M+4	LS for 16 bit register

Lets take a couple of examples. Let's load the B register with 195 and load the X register with $CØ98. Remember the $ sign means a hex value. If the memory starts at $3ØØØ then the values will be as follows:

Memory	Value	Function
$3ØØØ	$C6	Load B immediate
3ØØ1	195	Value to place in B
3ØØ2	$8E	Load X immediate
3ØØ3	$CØ	MS byte
3ØØ4	$98	LS byte

DIRECT ADDRESSING

The direct mode allows offset addressing with the value in the direct page register(DP) being the most significant byte. The direct page register is cleared or has a value of Ø after basic is initialized. Therefore memory locations in the first 256 locations of the memory map can be addressed using direct addressing with the DP=Ø. The format is the same as immediate addressing with the op code in the first memory location and the offset or value in the second location. Only two locations are required for either 8 or 16 bit registers. Exceptions are page 1 or page 2 instructions which will be discussed later.

EXTENDED ADDRESSING

This requires an op code plus two bytes for the address. This is similar to the way pointers or vectors are used in basic. For instance the start of basic is defined by the values in lo-

cations 25 and 26 with the most significant being in 25. So for extended addressing we will have:

M	OP Code
M+1	Most significant
M+2	Least significant

Extended addressing will require a total of 3 bytes. It covers the whole memory range.

INDEXED ADDRESSING

Perhaps this is the most powerful of the addressing modes. We can use this feature for quickly moving blocks of data or we can access data using a register as a reference and adding an offset. This allows us to write position independent code (PIC). A machine language program that is position independent will work in any free memory area.

USING X & Y REGISTERS

The X and Y registers can be used as pointers to give the address for loading and storing data. Remember that storing is similar to poking and loading is similar to getting a value by peeking in basic. The 6809 instructions allow a register to be loaded from a memory area designed by either the X or Y register and then to autoincrement the register. For example if X is pointing to 30000, we can load A indexed to X and autoincrement X. From this operation the value in 30000 will go into the A register, and X will increment to 30001.

Also we can store a value indexed to a register and increment the register. If we use the Y register to point to the location in which we want to store A, then Y=Y+1 after the operation.

MOVING DATA

This month we want to show how to move data using a machine language subroutine. Let's reserve the following memory locations for our pointers.

500-501 beginning of data
502-503 ending of data
504-505 new beginning

Let's write the step for our program.

(1) We will load the X register with 500.
(2) Load the Y register with 504.
(3) Load the A register indexed to X and increment X
(4) Store A indexed to Y
(5) Compare X with the value stored in 502 to see if we have finished. If we have not finished then go to (3)
(6) End or Return

MOVING DATA WITH BASIC

Let's show how data can be moved by using basic. We will use the variables A,X, and Y and write the program as similar as possible to the way we would write a machine language program.

```
10 'PROGRAM 9-1-86
20 'cOPYRIGHT (c) 1986
30 'dYNAMIC eLECTRONICS iNC.
40 'DEMONSTRATION PROGRAM FOR
    MOVING DATA USING BASIC
    THE BEGINNING VECTOR IS IN
    500-501, THE ENDING VECTOR
    IS IN 502-3, AND THE NEW
    BEGINNING LOCATION FOR THE
    DATA IS IN 504-5
50 'SET UP THE POINTERS X,Y,E
60 X=256*PEEK (500)+PEEK(501)
   'LDX EXTENDED WITH 500
70 Y=256 * PEEK (501)+PEEK(502)
   'LDY EXTENDED WITH 501
80 E=256 * PEEK(503) +PEEK(504)
   'THIS IS THE END OF DATA
90 A=PEEK(X):X=X+1 'LDDA INDEXE
   D WITH X AND AUTOINCREMENT X
100 POKE Y,A: Y=Y+1 'STA EXTEND
```

```
   ED INDEXED TO Y AND AUTOINCR
   EMENT Y
110 IF X<E THEN 50 'CMPX EXTEND
   ED WITH THE VECTOR IN 502 &
   GO TO 90 IF X IS LESS THAN E
120 END
```

The preceeding program is given to show how to move data. We will place the machine language subroutine at 510 as we have done in our previous examples. Let's look at the values and their function in each memory location. The * indicates the beginning of the instruction.

Memory	Value	Function
*510	190	LDX Extended
511	1	MS of 500
512	244	LS of 500
*513	16	Page 1 token
514	190	LDY Extended
515	1	MS of 504
516	248	LS of 504
*517	166	LDA Indexed to X
518	128	and Increment X
*519	167	STA Indexed to Y
520	160	and Incremnt Y
*521	188	CMPX Extended
522	1	MS of 502
523	246	LS of 502
*524	35	Branch if Less than or the same
525	247	to 517
*526	57	RTS

ML DATA MOVE PROGRAM

The following program loads the machine language subroutine from data statements and sets up the vectors to allow block moving of data.

```
10 ?"BLOCK MOVE ML PROGRAM
20 ?"cOPYRIGHT (c) 1986
30 ?"dYNAMIC eLECTRONICS iNC.
40 ?"PROGRAM 9-2-86
45 'SET UP THE VECTORS
50 INPUT "ENTER BEGINNING OF DA
   TA";X
60 M=INT(X/256):L=X-256*M:POKE
   500,M:POKE 501,L
70 INPUT "ENDING OF DATA";EN
80 M=INT(EN/256):L=EN-256*M:POKE
   502,M:POKE 503,L
90 INPUT "NEW LOCATION";Y
100 M=INT(Y/256):L=Y-256*M: POKE
   504,M:POKE 505,L
105 'READ IN THE ML DATA
110 FOR J=510 TO 526: READ A: PO
   KE J,A: NEXT J
120 EXEC 510
130 ?"THE DATA IS MOVED
140 DATA 190,1,244,16,190,1,248,1
    66,128,167,160,188,1,246,35,2
    47,57
```

Notice the power of this program. With only 17 bytes of machine language instructions, blocks of data can be moved anywhere within the memory map. Next month we will discuss more about indexed addressing and give additional examples. If you have a disassembler then you can disassemble the ML program after it is loaded.

SPECTROGRAM

MAGAZINE *FOR COLOR COMPUTER USERS.*

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

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

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

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

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

Name: _____

Address: _____

City: _____

State: _____ **Zip:** _____

( ) Check Enclosed ( ) Visa ( ) Mastercard

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

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

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

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

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

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

BEST

WE'VE CHOSEN THE BEST OF OVER 450 PROGRAMS AND PACKAGED THEM FOR YOU! 10 TO 12 PROGRAMS EACH PACKAGE. COLOR COMPUTER ONLY.

#1 Home Management I
Budget
Checkbook Balancer
Cost of Living
Tinycalc
Electronic Datebook
Account Manager
Stock Market
Word Processor
Lottery Analyst
Coco Database
Coco Terminal
Bartender

NEW!

#2 Home Management II
Video Cassette Organizer
Home Product Evaluation
Electric Gas & Water Cost
Baseball Manager
Car Manager
Ham Radio Log
Home Inventory
Personal Directory
Recipe Machine
Disk Labeler
Password Scrambler
Disk Directory Print

#3 Education
Flash Card
Spanish Lessons
Typing Tutor
Creativity Test
Arithmetic Football
Cost of Living
Math Tutors 1, 2
Trigonometry Tutor
Typing Game
Word Tests
Talking Alphabet
Clown Dunk Math

NEW!

#4 Adventures
Treasures of Barsoom
Killer Mansion
College Adventure
Coco Terrestrial
Escape
Zector
Skid Row
Quest
Naughide
Haunted House

#5 Games
Trek
Galactic Conquest
Warlords
The Power Sword
Steps
Robot Bomber
Force Field
Rat Attack
Caterpillar Cave
Meteor

#6 Utilities
Disk Dir Prot
Dir Pack & Sort
Disk Zapper
Roll Out
Doss Boss
Disk Backup
51/24 Editor
51/24 Screen
Autocopy
Funtsort
I/O Error Ignorer
Text Screen Print

Some of these programs above can sell for $29.95 each

$29.95 each
TAPE OR DISK

★★ SPECIAL ★★
BUY 2 PACKAGES AND GET THIRD ONE FREE

VISA

RAINBOW CERTIFICATION SEAL

THE GREATEST SOFTWARE DEAL ON EARTH!

GET 12 DISKS OR TAPES A YEAR CONTAINING OVER 120 QUALITY PROGRAMS. A SUBSCRIPTION TO T & D SOFTWARE CONSISTS OF 10 READY-TO-LOAD PROGRAMS DELIVERED BY FIRST CLASS MAIL EVERY MONTH.

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

PRICES

	TAPE OR DISK
1 YEAR (12 issues)	70.00
6 MO. (6 issues)	40.00
1 ISSUE	9.00

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

★ 16k-64k Color Computer
★ Over 3800 Satisfied Customers
★ Back Issues Available From
★ July '82 (Over 450 Programs)

OUR LATEST ISSUE CONTAINED
1 INCOME PROPERTY MANAGEMENT
2 BILL BOARD 2
3. MOUNTAIN BATTLE
4 THE TEN ROUND FIGHT
5 COCO - KEENO
6 HIGH RESOLUTION HOCKEY
7 LOGIC
8 ON SCREEN SCALE
9. LIBERTY SHIP
10 SINGLE STEP RUN

★ THIS MONTH ONLY ★

SUBSCRIBE FOR A YEAR AND RECEIVE A FREE PACKAGE OF YOUR CHOICE. INDICATE WHICH ONE
1. Home Man I
2. Home Man II
3. Education
4. Adventures
5. Games
6. Utilities

GET ISSUE #45 ABOVE FOR ONLY $3.00

RAINBOW CERTIFICATION SEAL

MasterCard

T & D SUBSCRIPTION SOFTWARE, P.O. BOX 256C, HOLLAND, MI 49423 (616) 396-7577

DISK FILE UTILITY

Have you ever wished you had a utility that would print all the information you needed in a chart or table? With this utility program three different tables of information can be printed.

Two directory tables and a file allocation table can be printed. We printed each of these for our disk containing our August files and have reduced and printed them here. The program contains instructions for using it and is provided by T & D Software. See their advertisement on page 7.

FILE ALLOCATION TABLE - TRACK 17, SECTOR 2

SIDE 1

AUG 86

FROM GRANULE	TO GRANULE	TRACK	FROM GRANULE	TO GRANULE	TRACK
Ø	FF	Ø	22	23	18
1	FF	Ø	23	24	18
2	FF	1	24	25	19
3	FF	1	25	C4	19
4	FF	2	26	27	2Ø
5	FF	2	27	C6	2Ø
6	FF	3	28	C8	21
7	FF	3	29	2A	21
8	FF	4	2A	2B	22
9	FF	4	2B	C3	22
A	FF	5	2C	2D	23
B	FF	5	2D	2E	23
C	D	6	2E	C2	24
D	C2	6	2F	C6	24
E	F	7	3Ø	31	25
F	C1	7	31	32	25
1Ø	11	8	32	33	26
11	C	8	33	C3	26
12	C2	9	34	FF	27
13	1Ø	9	35	FF	27
14	C3	1Ø	36	FF	28
15	12	1Ø	37	FF	28
16	17	11	38	FF	29
17	14	11	39	FF	29
18	C7	12	3A	FF	3Ø
19	C2	12	3B	FF	3Ø
1A	C3	13	3C	FF	31
1B	15	13	3D	FF	31
1C	C3	14	3E	FF	32
1D	1A	14	3F	FF	32
1E	C2	15	4Ø	FF	33
1F	1C	15	41	FF	33
2Ø	21	16	42	FF	34
21	C6	16	43	FF	34

DISK DIRECTORY

SIDE 1

AUG 86

FILENAME/EXT	TYPE	FMT	GR	START, END , EXEC	WRITE IN ANY OF YOUR COMMENTS HERE
HINTS /TXT	DAT	ASC	2		
MISC /TXT	DAT	ASC	4		
FRONT /TXT	DAT	ASC	1		
PG 2 /TXT	DAT	ASC	2		
HAM /TXT	DAT	ASC	2		
PG 6-8 /TXT	DAT	ASC	2		
ML PGM /BIN	ML	BIN	1	$308B,$37F6,$37F6	
8-2 /BIN	ML	BIN	3	$4146,$55D0,$55CF	
VIDEO /TXT	DAT	ASC	1		
SUB /BAS	BAS	ASC	1		
PG 4- /TXT	DAT	ASC	3		
8-5A /BIN	ML	BIN	3	$4146,$5460,$545F	
PG 11- /TXT	DAT	ASC	3		
PG 8-10 /TXT	DAT	ASC	4		
MORSE 3 /BAS	BAS	ASC	1		
DISK-DIS/BAS	BAS	ASC	5		
PG 14 /BIN	ML	BIN	2	$1E6E,$2852,$2852	

FREE GRANULES = 28

NUMBER OF PROGRAMS = 17

DIRECTORY - TRACK 17, SECTORS 3-11

SIDE 1

AUG 86

ENTRY	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
1	H	I	N	T	S				T	X	T	1	FF	20	0	87
2	M	I	S	C					T	X	T	1	FF	22	0	BA
3	F	R	O	N	T				T	X	T	1	FF	1E	0	C2
4	P	G		2					T	X	T	1	FF	1F	0	1C
5	H	A	M						T	X	T	1	FF	1D	0	F0
6	P	G		6	-	8			T	X	T	1	FF	26	0	E8
7	M	L		P	G	M			B	I	N	2	0	26	0	76
8	8	-	2						B	I	N	2	0	29	0	95
9	V	I	D	E	O				T	X	T	1	FF	18	0	4D
10	S	U	B						B	A	S	0	FF	19	0	D5
11	P	G		4	-				T	X	T	1	FF	16	0	BE
12	8	-	5	A					B	I	N	2	0	1B	0	25
13	P	G		1	1	-			T	X	T	1	FF	2C	0	AF
14	P	G		8	-	1	0		T	X	T	1	FF	30	0	C5
15	M	O	R	S	E		3		B	A	S	0	FF	2F	0	B5
16	D	I	S	K	-	D	I	S	B	A	S	0	FF	13	0	71
17	00	O	R	S	E		4	B	B	I	N	2	0	10	0	D9
18	P	G		1	4				B	I	N	2	0	E	0	EF

00-07 = FILENAME (IF 00 IS FIRST CHARACTER, FILE HAS BEEN DELETED.)
08-10 = EXTENSION
11 = FILE TYPE
(0=BASIC, 1=BASIC DATA, 2=MACHINE LANGUAGE, 3=TEXT EDITOR SOURCE)
12 = ASCII FLAG (0=BINARY, FF=ASCII)
13 = FIRST GRANULE NUMBER OF FILE
14-15 = BYTES USED IN LAST SECTOR
16-31 = UNUSED (FOR FUTURE USE)

DISK FILE
PROGRAM LISTING

```
1 REM COPYRIGHT (C) T&D SOFTWARE
    1985   * DISK UTIL *
10 CLEAR3000:DIMFA(67):FT$(0)="B
   AS":FT$(1)="DAT":FT$(2)="ML "
   :FT$(3)="EDT":AF$(0)="BIN":AF
   $(1)="ASC"
20 CLS:PRINT@3,"*** DISK FILE UT
   ILITY ***"
30 PRINT@72,"BY STEVE OSTROM"
40 PRINT@129,"FOR T&D SUBSCRIPTI
   ON SOFTWARE"
50 A$=INKEY$
60 PRINT@480,"INSTRUCTIONS (Y/N)
   ?";
70 A$=INKEY$:IFA$=""THEN70
80 IFA$="N"THEN250ELSEIFA$<>"Y"T
   HEN20
90 CLS:PRINT"THIS PROGRAM WILL A
   LLOW YOU TO  PRODUCE A PRINTE
   D COPY OF MOST  OF THE USEFUL
    DISK FILE DATA FOREITHER A S
   INGLE-SIDED DRIVE OR  ONE OR
   TWO DOUBLE-SIDED DRIVES."
100 PRINT:PRINT"FROM THE MENU YO
   U CAN CHOOSE ANYOF FIVE OPTIO
   NS:":PRINT:PRINT"1. SUPER DIR
   ECTORY                2. DIRECT
   ORY EXAMINE              3. FIL
   E ALLOCATION TABLE EXAMINE4.
   REPEAT THESE INSTRUCTIONS
   5. QUIT THE PROGRAM"
110 PRINT@480,"PRESS <ENTER> TO
   CONTINUE...";:INPUTA$
120 CLS:PRINT@4,"*** SUPER DIREC
   TORY ***"
130 PRINT@64,"THIS WILL GIVE YOU
    A PRINTOUT OFTHE DISK DIRECT
   ORY TO PLACE     INSIDE THE D
   ISK JACKET, AND WILLCONTAIN A
   LL THE FOLLOWING USEFULINFORM
   ATION:"
140 PRINT"FILE NAME AND EXTENSIO
   N, FILE   TYPE, FILE FORMAT,
   THE NUMBER OFGRANULES USED FO
   R FILE, START,  END AND EXECU
   TE ADDRESSES FOR   EACH MACHI
   NE LANGUAGE FILE, FREEGRANULE
   S LEFT ON DISK, NUMBER OFPROG
   RAMS ON DISK, AND A BLANK"
150 PRINT"COLUMN FOR YOUR COMMEN
   TS."
160 INPUT"PRESS <ENTER> TO CONTI
   NUE...";A$
170 CLS:PRINT@5,"*** DIRECTORY E
   XAMINE ***"
180 PRINT"THIS WILL GIVE YOU, IN
    TABULAR  FORM, A PRINTOUT OF
    ALL THE DATASTORED ON THE DI
   RECTORY TRACK,  INCLUDING FIL
   E NAME, EXTENSION, FILE TYPE,
    ASCII FLAG, NUMBER OFTHE FIR
   ST DISK GRANULE USED TO  STOR
   E THE PROGRAM AND THE NUMBER"
   ;
190 PRINT"OF BYTES USED IN THE L
   AST SECTOROF THE LAST GRANULE
    USED TO     STORE THE PROGRA
   M.  A PREVIOUSLYKILLED DIRECT
   ORY ENTRY WILL BE  SHOWN WITH
    ALL APPROPRIATE DATA,BUT UNU
   SED DIRECTORY LOCATIONS  WILL
    NOT BE SHOWN."
200 INPUT"PRESS <ENTER> TO CONTI
   NUE...";A$
210 CLS:PRINT@6,"*** FAT EXAMINE
    ***"
220 PRINT@64,"THIS WILL GIVE YOU
    A PRINTOUT OFTHE FILE ALLOCA
   TION TABLE (FAT) FOR THE DISK
   , FROM WHICH YOU CANTRACE THE
    LOCATIONS OF ALL THE  GRANUL
   ES USED TO STORE ANY FILE ON
   THE DISK.  THE FIRST COLUMN
   WILL INDICATE THE GRANULE NUM
   BER";
230 PRINT"AND THE SECOND COLUMN
   WILL SHOW WHICH GRANULE CONTA
   INS THE NEXT PART OF THE PROG
   RAM.  THE THIRD COLUMN SHOWS
   ON WHICH TRACK THE GRANULE WI
   LL BE LOCATED."
240 PRINT@480,"PRESS <ENTER> TO
   CONTINUE...";:INPUTA$
250 CLS:PRINT"        turn print
   er on":PRINT:PRINT"      inse
   rt desired disk":A$=INKEY$
260 PRINT:PRINT"DRIVE NUMBER (0-
   3) ?";
270 A$=INKEY$:IFA$=""THEN270
280 DR=ASC(A$):IFDR<48ORDR>51THE
   N270
290 DR=DR-48
300 CLS:DSKI$DR,17,11,X$,Y$:TI$=
   RIGHT$(Y$,8):PRINT"TITLE OF C
   URRENT DISK :"TI$:PRINT"DO YO
   U WISH TO RENAME (Y/N)":A$=IN
   KEY$
310 A$=INKEY$:IFA$="N"THEN340
320 IFA$<>"Y"THEN310
330 INPUT"NEW DISK TITLE (1-8 CH
   ARACTERS)";TI$:FORN=LEN(TI$)
   +1TO8:TI$=TI$+" ":NEXTN:Y$=LE
```

```
   FT$(Y$,12Ø)+TI$:DSKO$DR,17,11
   ,X$,Y$:GOTO3ØØ
34Ø CLS:PRINT@3,"*** DISK FILE U
   TILITY ***"
35Ø PRINT@11Ø,"MENU"
36Ø PRINT@16Ø,"1  SUPER DIRECTOR
   Y"
37Ø PRINT"2  DIRECTORY EXAMINE"
38Ø PRINT"3  FILE ALLOCATION TAB
   LE EXAMINE4  INSTRUCTIONS"
39Ø PRINT"5  QUIT"
4ØØ A$=INKEY$:SOUND1,1:LC=Ø
41Ø PRINT@49Ø,"YOUR CHOICE ?";
42Ø A$=INKEY$:IFA$<>""THEN47Ø
43Ø FORX=1TO1ØØ:NEXTX
44Ø PRINT@49Ø,"your choice";
45Ø FORX=1TO1ØØ:NEXTX
46Ø GOTO41Ø
47Ø PRINT@49Ø,"working       ";:O
   NVAL(A$)GOTO56Ø,91Ø,125Ø,9Ø,5
   5Ø
48Ø GOTO4ØØ
49Ø IFDR<2THENPRINT#-2,TAB(3Ø)"S
   IDE 1"ELSEPRINT#-2,TAB(3Ø)"SI
   DE 2"
5ØØ PRINT#-2:PRINT#-2,TAB(26);
51Ø PRINT#-2,CHR$(14);:REM PUT Y
   OUR PRINTER'S CODE HERE TO TU
   RN ON DOUBLE WIDTH PRINTING
52Ø PRINT#-2,TI$;
53Ø REM PUT YOUR PRINTER'S CODE
   HERE TO TURN OFF DOUBLE WIDTH
    PRINTING
54Ø PRINT#-2:PRINT#-2:PRINT#-2:R
   ETURN
55Ø CLS:END
56Ø PRINT#-2,TAB(26)"DISK DIRECT
   ORY":PRINT#-2:GOSUB49Ø
57Ø PRINT#-2,"FILENAME/EXT TYPE
   FMT GR  START, END , EXEC  WR
   ITE IN ANY OF YOUR COMMENTS H
   ERE"
58Ø PRINT#-2,"============ ====
   === ==  ================  ==
   ============================
   ===":PRINT#-2
59Ø DSKI$DR,17,2,X$,Y$
6ØØ FORI=ØTO67:FA(I)=ASC(MID$(X$
   ,I+1,1)):NEXTI
61Ø FORX=3TO11
62Ø DSKI$DR,17,X,X$,Y$
63Ø X$=X$+LEFT$(Y$,12Ø)
64Ø FORN=ØTO7
65Ø NA$=MID$(X$,N*32+1,8):EX$=MI
   D$(X$,N*32+9,3):GR=ASC(MID$(X
   $,N*32+14,1))
66Ø G1=GR:A=ASC(LEFT$(NA$,1))
67Ø FT$=MID$(X$,N*32+12,1):AF$=M
   ID$(X$,N*32+13,1)
68Ø IFA=255THENN=7:X=11:GOTO86Ø
69Ø IFA=ØTHEN86Ø
7ØØ AF=ASC(AF$)AND1:I=1
71Ø IFFA(GR)<128THENSL=GR:I=I+1:
   GR=FA(GR):GOTO71Ø
72Ø PRINT#-2,NA$+"/"+EX$;"  "FT$
   (ASC(FT$));" "AF$(AF);:IFI>9T
   HENPRINT#-2,I;:ELSEPRINT#-2,"
    "I;
73Ø IFASC(FT$)<>2THEN84Ø
74Ø LG=FA(GR):LS=LG AND31:LL=GR:
   LB=ASC(MID$(X$,N*32+16,1)):IF
   G1<34THENTN=INT(G1/2)ELSETN=I
   NT(G1/2)+1
75Ø SN=1+(G1 AND1)*9:DSKI$DR,TN,
   SN,A$,B$:SA=ASC(MID$(A$,4,1))
   *256+ASC(MID$(A$,5,1)):SA$=ST
   RING$(4-LEN(HEX$(SA)),"Ø")+HE
   X$(SA)
76Ø EA=SA+ASC(MID$(A$,2,1))*256+
   ASC(MID$(A$,3,1))-1:EA$=STRIN
   G$(4-LEN(HEX$(EA)),"Ø")+HEX$(
   EA):IFLL<34THENTN=INT(LL/2)EL
   SETN=INT(LL/2)+1
77Ø SN=(LL AND1)*9+LS:DSKI$DR,TN
   ,SN,A$,B$:A$=A$+LEFT$(B$,127)
   :IFLB=1THEN79ØELSEXA=ASC(MID$
   (A$,LB-1,1))*256+ASC(MID$(A$,
   LB,1))
78Ø XA$=STRING$(4-LEN(HEX$(XA)),
   "Ø")+HEX$(XA):GOTO83Ø
79Ø XA=ASC(MID$(A$,1,1)):IFLS=1T
   HEN81Ø
8ØØ DSKI$DR,TN,SN-1,A$,B$:XA=ASC
   (RIGHT$(B$,1))*256+XA:GOTO78Ø

81Ø IFSL<34THENTN=INT(SL/2)ELSET
   N=INT(SL/2)+1
82Ø SN=(SL AND1)*9+1Ø:GOTO8ØØ
83Ø PRINT#-2," $";SA$;",$";EA$;"
   ,$";XA$;
84Ø PRINT#-2
85Ø LC=LC+1
86Ø NEXTN,X
87Ø PRINT#-2:PRINT#-2,"FREE GRAN
   ULES =";FREE(DR)
88Ø PRINT#-2:PRINT#-2,"NUMBER OF
    PROGRAMS ="LC
89Ø NL=52:IFLC>52THENNL=118
9ØØ FORZ=1TONL-LC:PRINT#-2:NEXTZ
   :GOTO4ØØ
91Ø PRINT#-2,TAB(17)"DIRECTORY -
    TRACK 17, SECTORS 3-11":PRIN
   T#-2:GOSUB49Ø
92Ø PRINT#-2,"ENTRY   ØØ  Ø1  Ø2
    Ø3  Ø4  Ø5  Ø6  Ø7  Ø8  Ø9
    1Ø  11  12  13  14  15"
```

```
930 PRINT#-2,"=====  ";
940 FORX=0TO15
950 PRINT#-2," == ";
960 NEXTX
970 PRINT#-2:PRINT#-2
980 FORX=3TO11
990 DSKI$DR,17,X,A$,B$
1000 FORZ=1TO8
1010 ZZ=Z
1020 IFZ>4THENA$=B$:ZZ=Z-4
1030 W=0:A=ASC(MID$(A$,(ZZ-1)*32
   +1,1)):IFA=255THENZ=8:X=11:GO
   TO1120
1040 PRINT#-2,Z+(X-3)*8;
1050 IFA=0THENW=1:PRINT#-2,TAB(8
   )"00";
1060 FORY=W TO10
1070 PRINT#-2,TAB(Y*4+8);MID$(A$
   ,Y+(ZZ-1)*32+1,1);
1080 NEXTY
1090 FORY=11TO15
1100 PRINT#-2,TAB(Y*4+8);HEX$(AS
   C(MID$(A$,Y+(ZZ-1)*32+1,1)));

1110 NEXTY
1120 PRINT#-2:LC=LC+1
1130 NEXTZ
1140 NEXTX
1150 PRINT#-2:PRINT#-2:PRINT#-2:
   PRINT#-2,"00-07 = FILENAME (I
   F 00 IS FIRST CHARACTER, FILE
    HAS BEEN DELETED.)"
1160 PRINT#-2,"08-10 = EXTENSION
   "
1170 PRINT#-2,"   11 = FILE TYPE
   "
1180 PRINT#-2,"        (0=BASIC,
    1=BASIC DATA, 2=MACHINE LANG
   UAGE, 3=TEXT EDITOR SOURCE)"
1190 PRINT#-2,"   12 = ASCII FLA
   G (0=BINARY, FF=ASCII)"
1200 PRINT#-2,"   13 = FIRST GRA
   NULE NUMBER OF FILE"
1210 PRINT#-2,"14-15 = BYTES USE
   D IN LAST SECTOR"
1220 PRINT#-2,"16-31 = UNUSED (F
   OR FUTURE USE)"
1230 NL=45:IFLC>45THENNL=111
1240 FORX=1TONL-LC:PRINT#-2:NEXT
   X:GOTO400
1250 PRINT#-2,TAB(19)"FILE ALLOC
   ATION TABLE - TRACK 17, SECTO
   R 2":PRINT#-2:GOSUB490
1260 PRINT#-2,"FROM GRANULE   TO
    GRANULE   TRACK            F
   ROM GRANULE   TO GRANULE   TR
   ACK"
1270 PRINT#-2,"============   ==
   ========   =====            =
   ===========   ==========   ==
   ==="
1280 PRINT#-2
1290 DSKI$DR,17,2,A$,B$
1300 FORX=0TO33
1310 PRINT#-2,TAB(5)HEX$(X);TAB(
   19);HEX$(ASC(MID$(A$,X+1,1)))
   ;TAB(29)INT(X/2);TAB(50)HEX$(
   X+34);TAB(64);HEX$(ASC(MID$(A
   $,X+35,1)));TAB(74)INT(X/2)+1
   8
1320 NEXTX
1330 FORX=1TO22:PRINT#-2:NEXTX:G
   OTO400
```

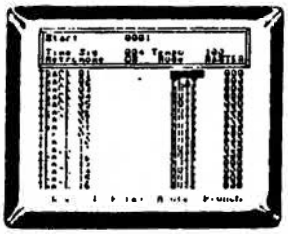
What Happens When You Own A
COLORCHESTRA™ MIDI SEQUENCER?

All Of A Sudden, Synthesized Music Production, And Recording Becomes Very, Very Simple.

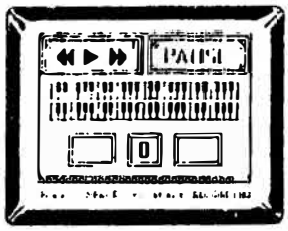
COLORCHESTRA, (from the author of CoCo MIDI), links together your Tandy 64K Color Computer and MIDI equipped keyboard synthesizer or rhythm drum machine and makes it simple to create masterpieces of music.

By incorporating menus and graphic icons, all there is to recording in real time is pushing a few keys.

A. Select track recording icon.



B. Select starting measure, time signature, tempo and recording track.



C. Simply play keyboard and hit break key when done.

THAT'S IT!

Once the track is entered, auto correction, transposing, and filtering may be implemented. And **COLORCHESTRA™** works with you to record up to **8,000 notes** utilizing as many as **16 tracks**...awesome.

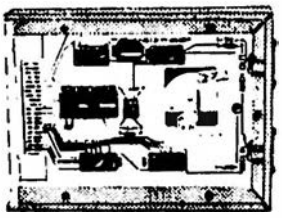
But it doesn't stop here - **COLORCHESTRA™** is crammed with a myriad of other outstanding professional features...

- ✔ Solo capabilities on any track
- ✔ Tempo range from 30-250 beats per minute
- ✔ Audible and visual metronome
- ✔ Programmable measure locator
- ✔ Sequencer will record from any MIDI Channel (1-16)
- ✔ Each track can output to any MIDI channel (1-16)
- ✔ Records full spectrum of MIDI data including program changes, pitch bends, all 128 MIDI controllers (modulation wheel, breath controller, sustain pedal, etc.)
- ✔ Will sync to drum machines
- ✔ MIDI thru on input
- ✔ Programmable time signature
- ✔ Real time velocity modification
- ✔ All 16 tracks can be filled
- ✔ Software filter removes specific MIDI parameters from recorded music such as pitch bend, program change, velocity data, modulation wheel, MIDI controller
- ✔ Transposition of notes up or down any number of octaves in half steps
- ✔ Auto correct feature for timing errors
- ✔ Stores composed music on tape or diskette
- ✔ Works with any disk operating system (Radio Shack, JDOS, ADOS, etc.)

THE COLORCHESTRA SYSTEM PACKAGE

HARDWARE

Encased between clear plastic panels and hand finished american walnut is **COLORCHESTRA's** sequencer board. Not just the edge connector, but every circuit trace is plated in 7 mil gold for optimum interface connection.



Two dependable, heavy duty 8' MIDI cables with metal jacketed end connectors are included.

SOFTWARE MEDIA

In addition to the hardware cartridge and cables, the **COLORCHESTRA™** System Package includes type set documentation in its own ring binder for easy reference and updating.

And for your convenience, both tape and diskette are supplied-so, if you need a back up it'll be there.

COLORCHESTRA™. A simple answer to your MIDI music production and HORIZON puts it all together for an introductory price of **149.95.....**

COLORCHESTRA™ system complete - $149.95 Call any day (ex. Sun.) to order. We ship same day. We accept check, COD, Visa, Master Card. Shipping add 3.00, COD add 2.00. Louisiana residents add 7 8 sales tax. Call for audio demonstration.

HORIZON
SOFTWARE CORPORATION

RAINBOW CERTIFICATION SEAL

318-942-1938 P O Box 289 Opelousas, Louisiana 70570

BASIC PROGRAMMING

A computer without instructions is not of much use. In this series we are showing how to write useful programs using Microsoft Basic. The methods presented here will work on all computers using Microsoft Basic.

PROGRAM TRACE

If a program does not work properly then it is necessary to determine the program steps that are causing the problem. If you have extended basic then you can use the trace commands. To turn the trace on enter the following:

TRON <ENTER>

With the trace enabled the number of the statement that is being processed will be printed on the screen with a [] surrounding the number. The screen can quickly become filled with numbers as the program goes from one line to the next. However if the computer is hanging up in a loop then this is a convenient way to observe the numbers and determine where the problem is occurring. The program can then be listed and steps taken to correct the problem.

To turn the trace off enter:

TROFF <ENTER>

Another method is to put labels in the program and print variables within suspected areas. The following examples demonstrate this:

```
20 ?"20 X$="X$
190 ?"190 J="J;"A(J)="A(J)
```

This methd is very useful because the variables are displayed along with the statement number. After the bugs are worked out of the program, lines 20 and 190 can be eliminated.

PROGRAM DEVELOPMENT

We have received many requests for programs for computers with 256K or 512K memories. What is unique about programs for these computers and why would special programs be needed? Last month we looked at a large address file containing 5000 names. If these were prospective customers and you wanted to mail literature to them, then the files would have to be arranged in order of assending or decending zip codes to take advantage of the low bulk mailing rate.

SETTING UP the FILES

```
1.James A. Smith (15)
2.1234 First St. (15)
3.Apartment 3A   (15)
4.Atlanta        (12)
5.GA             (2)
6.12345-9876     (10)
7.4047339884     (10)

79 bytes required
```

Figure 1

One way to set up a file would be to allot a fixed number of characters for each entry. An example is shown in Figure 1. This has the advantage of allowing complete files to be interchanged quickly. The disadvantage is that the number of bytes reserved has to be equal to the maximum number of characters for each entry. We can call this a fixed byte file. Let's take a look at a variable byte file.

Notice in Figure 2 the total amount of space required including one byte for a carriage return is 73. This is not much of a savings over the arrangement in Figure 1.

```
1.James A. Smith (15)
2.1234 First St. (15)
3.Apartment 3A   (13)
4.Atlanta        (8)
5.GA             (3)
6.12345-9876     (11)
7. 4047339884    (11)
```

Total bytes including carriage return = 76

Figure 2

Now consider the following:

```
1.Bill Jones     (11)
2.P. O. Box 123  (14)
3.               (Ø)
4.Danville       (9)
5.AL             (3)
6.35660          (6)
7.2055341983     (11)
```

53 Bytes

Figure 3

For short addresses such as post office boxes, the variable byte method would be much shorter. These examples are given to show the differences between a fixed byte file and a variable byte file. The fixed byte file has the advantage of allowing entire files to be quickly moved. This will make it easier to place them into an ordered pattern and will be the method we will use.

Let's again consider the organization of Figure 1 and write equations to determine the relative locations of each entry.

```
1.Smith,James A. (15) M+Ø
2.1234 First St. (15) M+14
3.Apartment 3A   (15) M+29
4.Atlanta        (12) M+41
5.GA             (2)  M+43
6.12345-9876     (1Ø) M+45
7. 4047339884    (1Ø) M+55
```

Total bytes required =79

Figure 4

Consider Figure 4. If M represents the memory location containing the first byte of information, then the equations at the right will allow us to quickly locate information. The numbers to the left will not be in the file but are placed in the figure for our benefit. We can possibly handle 3ØØ files in a 32K computer. For 3ØØ files we would need 3ØØ*79= 237ØØ bytes. This should leave us enough memory for our program plus random memory.

SORTING DATA

We have covered the procedure for entering information in our previous file program. How would we go about sorting the files? Suppose we want to do a bulk mailing and need to put the files in ascending zip codes. Let's look at the following collection of zip codes:

```
1. 32777
2. 98885
3. ØØ225
4. 4888Ø
5. 5233Ø
6. 11ØØ
7. 865ØØ
9. 932ØØ
```

Since we have a fixed byte file, we can quickly fine the zip codes within the files. How do we find the smallest. Look at the list of numbers and decide how you would make a new list using a pencil and paper. Fortunately we can do more advanced operations with the computer than by hand. Our sort procedure will be as follows:

1. Assume that the first number is the smallest.
2. Compare it with each number. If the first number is smaller then go to the next number. If the first number is larger then exchange the two numbers. So we will assume that 32777 is the smallest and compare it with

98885. 98885 is larger than 32777 so we will go to the next number. 32777 is less than ØØ225 so we will exchange the two numbers. We will continue this procedure until we reach the end of the file. The first number will be the smallest.

Next we will compare the second number with the rest and exchange them until we reach the end of the file. Then the first number will be the smallest and the second number will be next to the smallest. We will repeat this procedure until the numbers are in assending order.

With machine language subroutines we can quickly exchange the entire files associated with the zip codes. When we finish the sort routine then the files will be in order.

Files with a Disk or Ramdisk

For a large number of addresses, it will be necessary to pull files from a disk or ramdisk. We may want to rearrange all of the files on the disk so that the data will be in a desired order. So our control program will have to be oriented toward not only handling the data within the computer, but also merging and modifying other files on disk or ramdisk. We will continue with this next month.

HAM RADIO & COMPUTERS

by

Bill Chapple W4GQC

Computers are wonderful tools for doing numerous tasks. They can be used for storing data and information and for controlling devices. There are many applications for using computers in ham radio. Most of the ham radio hardware and software is for Comodore computers. The Radio Shack Color Computers are easy to adapt to ham radio applications. Last month we gave a Morse Code program. One of the requirements for obtaining an FCC licence is to be able to copy the International Morse code. There are many other applications that I want to cover and some involve hardware. For example the computer can be made to receive Morse code and display the characters. This will require a hardware interface and possibly a machine language subroutine. An interface can be built so that the computer can generate Morse code and key the transmitter. This will be coming soon. Interfacing computers is one of our current series and we will be presenting some hardware adapters for these purposes.

CALCULATING ANTENNAS

A radio station is no better than its antenna. I have had much experience in building antennas for the frequencies below 3ØMHZ. Antennas can be cut from formulas, assembled, and installed without test equipment. I have a grid-dip meter which can tell me if an antenna is resonant. I also have an impedance bridge that I built in 196Ø. An impedance bridge gives the impedance of the antenna. To get maximum power transferred to the antenna, the transmission lines' impedance must match the impedance of the antenna.

The impedance of a dipole is around 5Ø ohms. RG-8 or RG-58 cable has an impedance of 5Ø ohms and can be used for the transmission line. For some of our readers who might not be familiar with this cable but are familiar with citizens band radio, RG-58U is the type of cable that runs from the antenna to the CB radio.

The material for the elements can be copper wire or aluminum tubing. The element to which the transmission line connects is called the driven element (DE). Our program will calculate its length. It needs to be cut into 2 pieces and the transmission line fed at the center with the braid or outter conductor connected to one piece and the center conducter connected to the other piece. The longest element is the reflector (RE). The shorter elements are called directors (D1 & D2). Maximum radiation is from the reflector towards the director(s).

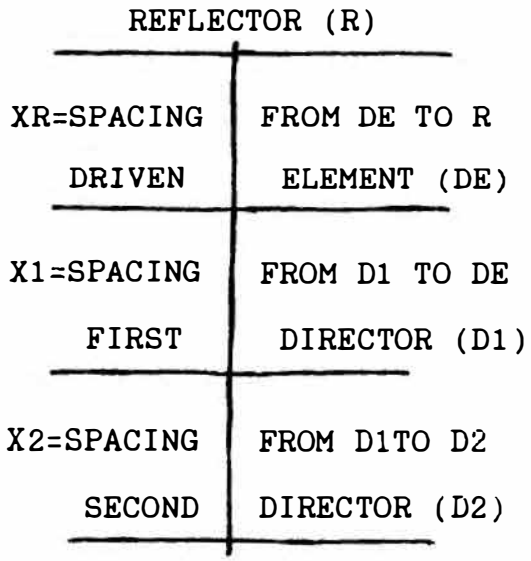
We suggest you refer to an antenna or the ARRL handbook for additional information on antenna fabrication and installation. The equations we are presenting work for CB radio, Television, FM radio, Business radios, etc.

For our program we are reserving the following variables:

```
RE=REFLECTOR LENGTH
DE=DRIVEN ELEMENT LENGTH
D1=FIRST DIRECTOR LENGTH
D2=SECOND DIRECTOR LENGTH
XR=SPACING FROM DE TO RE
X1=SPACING FROM DE TO D1
X2=SPACING FROM D1 TO D2
```

A sketch of an antenna up to 4 elements is shown in figure 1. Maximum radiation is from the reflector to the director. The antenna can be mounted either horizontally or vertically. The results are in feet and inches. The inches are in whole inches and decimals. You

can round the decimal up or down to the nearest inch for frequencies below 30 MHZ. The decimal can be multiplied by .8 to give eights of an inch.



Antenna Configuration

Figure 1

ANTENNA DESIGN
PROGRAM LISTING

```
10 CLS:PRINT"    ANTENNA DESIGN
   PROGRAM" ,"cOPYRIGHT (c) 1986
   ","dYNAMIC eLECTRONICS iNC.",
   "PROGRAM 9-2-86
20 PRINT"THIS IS AN ANTENNA PROG
   RAM THAT","IS DESIGNED FOR UP
    TO 4 WIDE","SPACED ELEMENTS.
    THE RESULTS AREGIVEN IN FEET
    AND INCHES":GOSUB 170
30 PRINT"THESE ARE WIDE SPACED E
   LEMENTS","AND LOADING IS NOT
   A PROBLEM","WHEN ADDITIONAL E
   LEMENTS ARE","ADDED. USE THE
   CALCULATIONS FOR","A DIPOLE,
   OR ANY COMBINATION OF","ELEME
   NTS. ADD THE REFLECTOR","AND
   THEN THE DIRECTORS FOR","LARG
   ER ARRAYS."
40 GOSUB 170:CLS:INPUT"ENTER FRE
   QUENCY IN MEGAHERTZ";F
50 DE=475/F:W=DE/.47:RE=.51*W
60 D1=.45*W:D2=.44*W:XR=.2*W:X1=
   .2*W:X2=.25*W
70 P$="##
80 CLS:PRINT"FREQUENCY="F
90 PRINT"REFLECTOR LENGTH=";:V=R
   E:GOSUB 180
100 PRINT"DRIVEN ELEMENT=";:V=DE
   :GOSUB 180
110 PRINT"FIRST DIRECTOR=";:V=D1
   :GOSUB 180
120 PRINT"SECOND DIRECTOR=";:V=D
   2:GOSUB 180
130 PRINT"DE TO R=";:V=XR:GOSUB
   180
140 PRINT"DE TO D1=";:V=X1:GOSUB
    180
150 PRINT"D1 TO D2=";:V=X2:GOSUB
    180
160 GOTO 40
170 INPUT"PRESS ENTER TO CONTINU
   E";Q:CLS :RETURN
180 Y=INT(V):Z=(V-Y)*12:PRINTY;"
   FT ";:PRINTUSING "##.#";Z;:PR
   INT" IN":RETURN
```

INTERFACING COMPUTERS
(Part 8)

In this series we are looking at ways of getting information from the computer and putting information into the computer through external devices. Color Computers have 2 joystick ports, a serial ASCII port, and a parallel or expansion port.

For the past few months we have been looking at using the serial ASCII port for sending and receiving characters or data. We have explained in detail how data is formatted for serial use. As a review a logical "1" is on the output until a character is to be sent. Then a start bit which is a "Ø" is sent to indicate the start of the timing sequence. Next the 7 or 8 data bits are sent, followed by a parity bit and one or two stop bits.

We are presenting material for developing a terminal program that will allow a color computer to exchange information with and other device using a RS-232 port. This month we want to develop the machine language portion that allows characters or data to be sent from the computer to another device. The most important part will be the timing subroutine.

TIMING SUBROUTINE

Memory locations 149 & 15Ø contain the baud rate used for sending data to a printer. We can use this for our machine language timing subroutine. One way to provide a timing delay is to load the A & B registers with values and decrement them until the values are Ø. The following is a position independent routine for a delay:

```
M    LDA Immediate with 3
M+2  LDB Direct with 15Ø
M+4  DECB  'B=B-1
M+5  BNE M+4 'GO TO M+4 IF =Ø
M+7  DECA  'A=A-1
M+8  BNE M+2 'GO TO M+2 IF =Ø
M+1Ø RTS
```

OUTPUT LOCATION

To cause the output to change it is necessary to store or poke a value into address location 65312. Poking a Ø into this location causes a Ø to appear on the output. The second least significant bit actually controls the output. We can put the proper bit in this location by using one of the assembly rotate or arithmetic shift commands. The rotate command moves all bits of a byte one location either to the left or right. The shift command also moves the bits to the right or left but retains the bit in the first location. Refer to an assembly language book or the 68Ø9 data sheet for more information. Let's give an example of rotating a byte.

B7	B6	B5	B4	B3	B2	B1	BØ	
1	Ø	Ø	1	1	Ø	Ø	1	Byte
Ø	Ø	1	1	Ø	Ø	1	C	Rotate left

For rotating a byte the bit in B7 goes to the carry location of the conditional code register and the carry bit goes into BØ. All of the other bits are shifted left one location.

If we want to output the bit in BØ then we can rotate the byte left (ROL) in the A register and store it in 65312.

REMOVING the BITS from a BYTE

One approach is to store the byte in a memory location. Then load the A register with the byte and AND A immediate with Ø1. This leaves the least significant bit. Then we can arithmetic shift left the A register and store the register

in 65312 to output the bit. Next we can rotate right the byte stored in memory so that the next bit will be in the least significant (BØ) location. We also need a bit counter so that we will know when we have finished. This can be stored in a memory location and decremented each time we output a bit until we have finished. We can then output the parity bit and the stop bits to complete the character.

OUTPUT ML SUBROUTINE

This collection of subroutines is used on our DYTERM terminal program. We will use it in for our terminal development program here. It is written in position independent code which means that it will work in any memory location. If you have an assembler then you can assemble the program or poke the values into memory and use your disassembler. We used our decimal assembler "DISASM" to disassemble this so all branch locations will be in decimal. We use the following symbols for addressing modes:

I - Immediate
D - Direct
E - Extended
N - Indexed

ML SUBROUTINES
for SENDING an ASCII BYTE

Memory	Value -Hex	Function
	Time Subroutine	
ØØ	134-86	LDA I 3 'Put a 3
Ø1	3	into A
Ø2	214-D6	LDB D 15Ø 'B=
Ø3	15Ø-96	PEEK (15Ø)
Ø4	9Ø-5A	DECB 'B=B-1
Ø5	38-26	BNE Ø4 'GO TO 4
Ø6	253-FD	IF NOT =
Ø7	74-4A	DECA 'A=A-1
Ø8	38-26	BNE Ø2
Ø9	248-F8	
1Ø	57-39	RTS 'RETURN
11	182-B6	LDA E 4ØØ5 'A=
12	15-ØF	PEEK(4ØØ5)
13	165-A5	WORD LENGTH
14	183-B7	STA E 4ØØ9 '
15	15-ØF	POKE 4ØØ9,A
16	169-A9	
17	127-7F	CLR E 65312'
18	255-FF	SEND START
19	32-2Ø	BIT
2Ø	141-8D	BSR Ø 'GO TO
21	234-EA	TIME SUB
22	79-4F	CLRA
23	183-B7	STA E 4ØØ7
24	15-ØF	
25	167-A7	
26	182-B6	LDA E 4Ø1Ø
27	15-ØF	
28	17Ø-AA	
29	132-84	ANDA I 1'
3Ø	1	A=A AND 1
31	39-27	BEQ 36
32	3	
33	124-7C	INC E 4ØØ7'
34	15-ØF	M=M+1
35	167-A7	COUNTER
36	72-48	ASLA
37	183-B7	STA E 65312'
38	255-FF	OUTPUT BIT
39	32-2Ø	
4Ø	122-7A	DEC E 4ØØ9 '
41	15-ØF	M=M-1
42	169-A9	
43	118-76	ROR E 4Ø1Ø
44	15-ØF	'ROTATE RIGHT
45	17Ø-AA	4Ø1Ø
46	141-8D	BSR Ø 'BRANCH
47	2Ø8-DØ	TO TIME SUB
48	125-7D	TST E 4ØØ9'
49	15-ØF	IS 4ØØ9=Ø?
5Ø	169-A9	
51	38-26	BNE 26 'GET
52	229-E5	NEXT BIT
53	246-F6	LDB E 4ØØ8 '
54	15-ØF	GET PARITY
55	168-A8	
56	182-B6	LDA E 4ØØ7 '
57	15-ØF	BIT COUNTER
58	167-A7	
59	132-84	ANDA I Ø1
6Ø	1	
61	93-5D	TST B 'IS B=Ø?
62	39-27	BEQ 77 'GO TO
63	13-ØD	77 IF B=Ø
64	193-C1	CMPB I 1 '

NEW 256K MEMORIES

for 2-chip CC-2 (ME-18)
16K or 64K to 256K

Have you ever wished you could stop what you are doing, load another program, and then return to the original program without loosing anything? This is possible with our new ME-18 expanders. This plug in assembly increases the memory 4 times. The memory assembly is in two modules partitioned as 4-64K memory banks which are hardware selectable by two toggle switches. Features include:

* Powerful Memory Manager Software to allow maximum use of each 64K bank.

* 4-64K memories. You can load any combination of 64K programs such as word processors, OS-9, terminal programs, or spread sheets. Each bank is entirely independent allowing you to quickly go from one to the other by selecting the bank with the toggle switch.

* Ramdisk in each bank. Basic or machine language programs can be stored in the second 32K bank for any of the selected 64K memory banks. You can have special programs in one or two banks and your basic programs in the other banks. The ramdisk quickly loads and runs the programs from the computer's memory.

* Independent banks. Each of the 4 banks is completely independent allowing any combination of programs to be entered. The unselected banks are protected and the data can not be altered until the bank is again selected. For example one bank can contain a word processor, the second a machine language game program, the third a terminal program, and the fourth a spread sheet. When banks are switched all variables are preserved allowing the program to run or continued when the banks are reselected.

* Plug in installation. For 64K computers, installation involves removing the two memory chips and inserting the assemblies into the empty sockets. Two small holes are required for the switches to complete the installation. For 16K computers a jumper must be soldered to upgrade the computer to 256K.

* Low cost. ME-18 $119.95

128K UPGRADES

ME-10A Similar to the ME-18 except upgrades 2-chip 64K computers to 128K for 2-64K bank operation. Ramdisk software is included. $49.95

ME-12 Upgrades 8-chip 4164 type 64K computers to 128K. Ramdisk software is included. $49.95.

64K UPGRADE

ME-10 Upgrades 16K CC-2 to 64K. Ramdisk software is included $34.95.

EXTENDED BASIC

Add extended basic to CC-2 computers $34.95.

Free Catalog

24 Hour phone. Checks, VISA & MC cards. Add $3 ship.

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

```
  65      1       IS B=1?
66       39-27   BEQ 69 'GO TO
  67      3       69 IF B=1
68       67-43   COMA 'COMPLEMENT A
69      132-84   ANDA I 1 'A=A
  7Ø      1       AND 1
71       72-48   ASLA 'SHIFT LEFT
72      183-B7   STA E 65312 '
  73    255-FF    OUTPUT BIT
  74     32-2Ø
75      141-8D   BSR Ø 'TIME SUB
  76    179-B3
77      182-B6   LDA E 4ØØ5
  78     15-ØF    'GET NUMBER OF
  79    165-A5     STOP BITS
8Ø      183-B7   STA E 4ØØ4
  81     15-ØF    'SAVE NUMBER
  82    164-A4
83      134-86   LDA I 255
  84    255-FF    'A=255
85      183-B7   STA E 65312
  86    255-FF    'OUTPUT "1"
  87     32-2Ø
88      141-8D   BSR Ø 'TIME SUB
  89    166-A6
9Ø      122-7A   DEC E 4ØØ4
  91     15-ØF    M=M-1
  92    164-A4
93      125-7D   TST E 4ØØ4
  94     15-ØF   'HAVE WE
  95    164-A4    FINISHED?
96       38-26   BNE 83
  97    241-F1
98       57-39   RTS 'RETURN
```

If you are interested in developing your own terminal program we suggest you write a basic program and enter the decimal values into data statements. Then you can the data and poke the values into memory. Or you can poke them into memory and save the data as a ML program. For example if you place the data starting at 4100 then enter:

```
(C)SAVEM "DATA", 4100, 4200, 4100
```

Next month we will give subroutines for receiving characters from another computer. We can use these subroutines as I/O links from a basic control program.

EDITOR'S COMMENTS

We are very excited about the new Color Computer 3. However we have not seen one yet and everytime I check at a Radio Shack Store they say they are being expected any day now. So I guess we will keep checking and they will arrive someday.

While looking back over the articles we have written, I noticed that we are very short on hardware articles. So next month we will have a hardware article. I have thought about many hardware projects and almost all of them involve software. For example in our interfacing series, we are developing software for a terminal program. This can be very useful for interfacing with other devices because no modifications to the computer are required.

A project in which I have been interested for quite some time is a computer controlled sign. With this sign we could list items we have for sale for local customers. We had a T G & Y department store here which recently closed. They kept reducing prices until everything was sold. I bought two boxes of Christmas tree light bulbs which could be used for my sign. Of course it will take some dedicated electronics to control the bulbs, but a Color Computer could be used to control the messages. I will keep you informed with our progress on this project.

We have had some positive response to our Ham Radio series. Of course we can use the ASCII port for interfacing with radio equipment. One thing that has always facinated me about computers is their adaptability. This month we have a program for calculating antennas.

Next month we will start a series on OS-9. We have had many requests for this so we

will begin next month. If you have a subject you would like for us to cover please let us know.

We also need programs. If you have a program that has not been published send it to us and we will make you an offer. We have received a few but could use more.

We are looking at IBM compatible computers. These are expandable and there is much software available for them. There are some things that are easier to do on a Color Computer and we might make some comparisions in a the future. We have a Radio Shack model 100 on which we write many of our editorials. We wrote a program to transfer files from the model 100 to a Color Computer. Using the ASCII port we can do the same with an IBM compatible or any other computer that has a RS-232 port.

PRODUCT REVIEWS

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

TX WORD PROCESSOR

TX is a printer page editor. It requires a 64K computer and a disk drive. A printer page has 80 x 66 or 5200 locations. The standard 32 x 16 screen is a window that can be moved anywhere within the page.

When the program is run a menu appears. From the menu you can go to the the screen buffer where you can write characters. Other options are erasing the screen, saving a screen or block buffer, loading a file into the screen or block buffer.

Text Mode

In the text mode the cursor indicates the position into which a character will be written. If a character is in the location indicated by the cursor, and a new character is entered, the new character replaces the old character. The cursor can be moved by the 4 arrow keys. Pressing the shift and then an arrow key moves the cursor 8 locations. The number of the row and column is displayed at the bottom of the screen. A shift "." moves all characters starting under the cursor one location to the right. A shift "," moves all characters one location to the left.

Clear Mode

This mode provides additional features. Pressing "C" centers the text. A question mark appears and the number of lines to be centered should be entered. Pressing "R" allows the right margin to be reset to any number between 40 and 80. A character can be repeated vertically by pressing "V" and entering the number of characters. Pressing "S" allows tabs to be set and "K" kills a line of text.

Blocks of characters can be moved to any location. The beginning and ending of the block is marked. Then it is placed into the block buffer. Move the cursor to a new location, press "CLEAR P" and a copy of the block will be printed at the cursor's location. Blocks can be erased by a similar procedure.

The total for a column of numbers can be calculated. The beginning of the numbers and ending has to be marked. Then pressing "CLEAR T" prints the result under the column. Other

features include inserting printer codes at the beginning of a line and line swapping.

Summary

We found TX to be very useful. Since it is a page editor its greatest use would be for applications that only require one page at a time. The column calculator feature would be handy for applications such as balance sheets or monthly sales reports. The program sells for $16.95 plus $2 S/H. For more information contact Fred Kolesar, 7 Ladd Road, Westfield, PA 16950 (814) 367-5384.

+ + + DCN STAFF + + +

NEW PRODUCTS

This section is available free for producers and dealers of color computer products. These products have not been reviewed by us but are included for our reader's information.

NEW 256K MEMORIES

Dynamic Electronics Inc. has designed a 256K memory upgrade for the newer CoCO 2's with two memory chips. The memory assembly, designed ME=18, is in two modules partitioned as 4-64K memory banks which are hardware selectable by two toggle switches.

Memory manager software is included to allow maximum use of each 64K bank. With this software the second 32K of memory in each 64K bank can be used for either programs or a ramdisk. If programs are stored in each 32K bank then any of the 8 programs can be quickly run.

Each of the 4 banks is completely independent allowing any combination of programs to be entered. The unselected banks are protected and the data can not be altered until the bank is again selected. For example one bank can contain a word processor, the second a machine language game program, the third a terminal program, and the fourth a spread sheet. When banks are switched all variables are preserved allowing the program to run or continue when the banks are reselected.

For 64K computers, installation involves removing the two memory chips and inserting the assemblies into the empty sockets. Two small holes are required for the switches to complete the installation. For 16K computers a jumper must bes soldered to upgrade the computer to 256K.

Features include plug in installation, memory protection for the unselected banks, and memory manager software. Cost $119.95 + $3 S/H. For additional information contact Dynamic Electronics Inc., P. O. Box 896, Hartselle, AL 35640.

OPERATING HINT

You can print your disk directory to a printer by POKE 111,254:DIR <ENTER>

OPERATING HINT

Checking Tape Programs - You can check the programs on a cassette tape by using the SKIPF command. Load the tape and rewind it. Then type SKIPF"X where X is a file that is not on the tape. The name of each file will be displayed on the screen as they are found on the tape. If there is an error the computer will give an error message and stop the recorder. All files or programs before the recorder stopped are good. If the recorder goes to the end of the tape without indicating an error then all of the files are good. Press the rear reset button to reset the computer.

QUESTIONS & ANSWERS

These are questions that have been asked us. If you have a computer question please write and we will answer it here. For a quick reply send $10 with your question.

QUESTION: I am using a RS Color Computer with a program for packet radio. The program calls for a CONTROL Function. Is this available on the CoCo?

ANSWER: The Radio Shack Color Computers do not have control keys. A control key subtracts 64 from the ASCII value of the pressed key. You can use a key to set up the control function. The folowing is an example program.

```
10 X$=INKEY$: IF X$="" THEN 10
20 A=ASC(X$): IF A=94 THEN 90
30 ?X$;:?#-2,X$;
40 GO TO 10
80 'ENTER CONTROL CHARACTER
90 Y$=INKEY$: IF Y$="" THEN 90
100 Y=ASC(Y$): Y=Y-64: IF Y<1
    THEN Y=0
110 ?#-2,CHR$(Y);:GOTO 10
```

RENEWAL TIME?

The date beside your name on the address label indcates the last issue you will receive. Send in your renewal if you want to continue receiving technical information on Color Computers. This is the last issue for those with **9/86**.

DISPLAY ADS

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

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

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

We can do ads in Red, Blue, or Brown. No all one color ads will be accepted. For color ads send artwork for each color. Add 4Ø% for each color. Example: One page black and red for 3 times costs $25 + 1Ø.ØØ = $35.ØØ each month.

DCN PROGRAMS on Tape or DISK

This is our third collection of programs from Dynamic Color News. This collection includes:

1. RESTORE - Page -1 Program that restores a basic pgm which was lost due to a hard reset or typing NEW.

2. FAST FOOD - This program quickly displays the total for a fast food order.

3. BAR GRAPH - Display results in easy to see bars over a 12 month period.

4. MEMORY PEEK & POKE - Page -1 program that can be loaded with another pgm.

5. GRAPHICS DRAW. Draw figures on the screen. Save and load drawings.

DCN-3 Tape or Disk $11.95
Add $2 shipping, Foreign $3

Please sign me up for one year for the DYNAMIC COLOR NEWS. I want to receive instruction on programming, Computer Theory, Operating Techniques, Computer Expansion, plus information on New Products, and Product Reviews. I understand that there will be no charge for answers to questions printed in the Newsletter.

Cost $15 USA & Canada, $3Ø foreign.

Name _____ Mail payment to
Address _____ Dynamic Electronics Inc
City _____ P. O. Box 896
State & Zip _____ Hartselle, AL 35640
Enclosed is a check ___
charge to VISA ___ MC ___ Number _____Exp._____

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