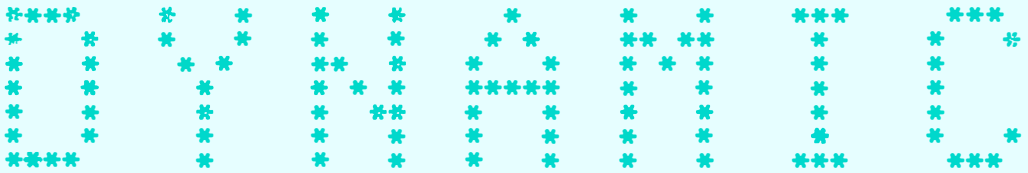
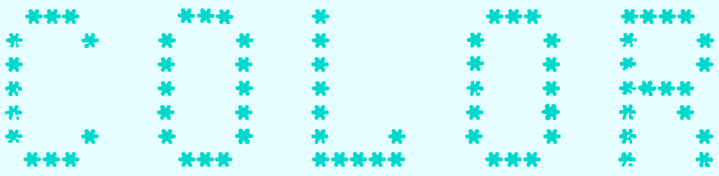
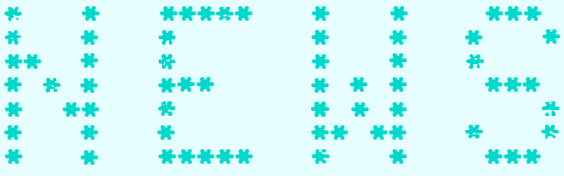
ENGINEERING NOTES
on
Radio Shack Color Computers

January 1985 $1.95
Vol. 1 No. 11







* RANDOM NUMBERS (Part 2)
* CARD SHUFFLING PROGRAM

* COMPUTER SOUNDS (Part 3)
* BASIC PROGRAMMING
* SOUND DEMONSTRATION PROGRAM
* MACHINE LANGUAGE PROGRAMMING
* OPERATING HINTS
* QUESTIONS & ANSWERS

DYNAMIC COLOR NEWS is published monthly by DYNAMIC ELECTRONICS, INC., P.O. Box 896, Hartselle, AL 35640, phone (205) 773-2758. Bill Chapple, President; Alene 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

January 1985

Editor and Publisher
Bill Chapple

Secretary
Helen Earls

CONTENTS

Editor's Comments 3

Random Numbers 3
Part 2

Card Shuffling 3
Program Listing

Shuffling Program 4

Computer Sounds (Part 3) . . 5

Machine Language Sub. . . . 6

Sound Program Discussion . . 6

Operating Hints 8

Questions & Answers 8

Next Month 8

EDITOR'S COMMENTS

This isssue will mark the end of our first year of writing this newsletter. The experience has been very exciting for us and we hope that you have learned a great deal from having read our editorials. There is always more that could be done and as we approach our second year we want to implement more things for the newsletter. We want to do Product Reviews, and have a news section on New Products. We will also be taking advertising from other firms to help defray our costs.

This month we are completing our series on Computer generated sounds. We included a Basic program with Machine Language Subroutines that will allow you to hear the sounds that are generated by 3 popular waveforms. Last month we introduced Random Numbers and this month we have a card shuffling and dealing program that uses random numbers.

Because of the number of requests for information on writing programs to use all 64K in 64K computers and to be used on 128K computers, we are starting a series of editorials on this subject. When these larger memories are available then special program writing techniques are required. We are going to present some of these techniques so that you can write programs for these larger memories.

Next month we will have a hardware project you can build for your computer. Would you like to see more of these? We would like to hear from you concerning subjects you would want us to discuss in the newsletter.

As the new year is upon us we want to wish each of you a Happy New Year and a very successful 1985.

RANDOM NUMBERS
Part 2

Last month we introduced random numbers and showed how they could be generated from Basic. We showed how to generate random sounds and discussed the problem of shuffling a deck of playing cards. This month we wrote a program that will shuffle a deck of cards and deal them. We didn't put them in order but if you desire this you can use the sort technique that we presented in an earlier issue.

We mentioned last month two approaches to the problem of sorting a deck of cards. The first approach would be to pick one of the 4 suits and then to pick a card from that suit. There are 4 suits and 13 cards in each suit. The second approach would be to number the cards from 1 to 52 and randomly pick cards. This is the approach we took because it seemed less confusing.

CARD SHUFFLING
Program Listing

```
2 ?"THIS PROGRAM SHUFFLES CARDS
4 ?"   cOPYRIGHT (c) 1984
6 ?"  dYNAMIC eLECTRONICS iNC.
10 ?"    THIS PGM IS RND-1
15 ?
20 DIM N(13),S(13),E(13),W(13),
   D(52),X(52),S$(52)
30 FOR J=1 TO 52
35 READ S$(J)
40 X(J)=0:D(J)=J
50 NEXT J
60 PRINT"THIS SHUFFLES THE DECK
70 FOR K=1 TO 52
80 A=RND(52)
90 IF A=0 THEN 80
100 IF D(A)=0 THEN 80
110 X(K)=A: D(A)=0
120 NEXT K
130 FOR K=1 TO 52:PRINT K;X(K);
    S$(X(K))
150 NEXT K
```

```
160 PRINT"THIS DEALS THE DECK
165 FOR X=1 TO 1000:NEXT X
170 PRINT"WEST   NORTH  EAST
    SOUTH
180 FOR J=1 TO 13
190 FOR K=0 TO 3
200 V=4*J+K-3
210 W(J)=X(V):N(J)=X(V+1)
220 E(J)=X(V+2):S(J)=X(V+3)
230 NEXT K
235 PRINTS$(N(J))"      ";
    S$(S(J))"      ";S$(E(J));
    "      ";S$(W(J))
240 NEXT J
255 DATA 2*,3*,4*,5*,6*,7*,8*,
    9*,10*,J*,Q*,K*,A*
265 DATA 2<,3<,4<,5<,6<,7<,8<,
    9<,10<,J<,Q<,K<,A<
275 DATA 2&,3&,4&,5&,6&,7&,8&,
    9&,10&,J&,Q&,K&,A&
285 DATA 2@,3@,4@,5@,6@,7@,8@,
    9@,10@,J@,Q@,K@,A@
```

SHUFFLING PROGRAM

(See the Shuffling Program Listing)

The first thing we did was to set up arrays for the cards. We used the dimension statement and set up an array for each hand which we called N(13), S(13, E(13), W(13) for the North, South, East, and West hands. Also we set up arrays D(52) for the deck, X(52) for the shuffled deck, and S$(52) for the names of the cards. This is in statement 20. Statements 30 - 50 initializes the arrays. Notice we used the familiar Read and Data method of handling the names of the cards.

Shuffling Routine

Statements 60 through 130 shuffles the deck and prints the results. Let's discuss our shuffling technique. We have two arrays representing the original deck and the new deck. These are labeled D(N) and X(N). In our initializaton we put the number of the card in the corresponding location for that card in the original deck. In other words D(31) = 31 or D(N)= N. As we randomly select the cards we will replace the value for the card selected with a zero. On our next pass if we selected a card that had previously been selected then we will know this because the value will be zero. This is A in statements 80 and 90. The random number is selected in 80 and tested to see if the value in that location of the array is zero. This is statement 100. If the value (A) is zero then that number has already been selected and we need to go back to 80 and pick another value. Using this approach it takes a few seconds to shuffle the deck. Statements 130 - 150 prints the results of the shuffled deck.

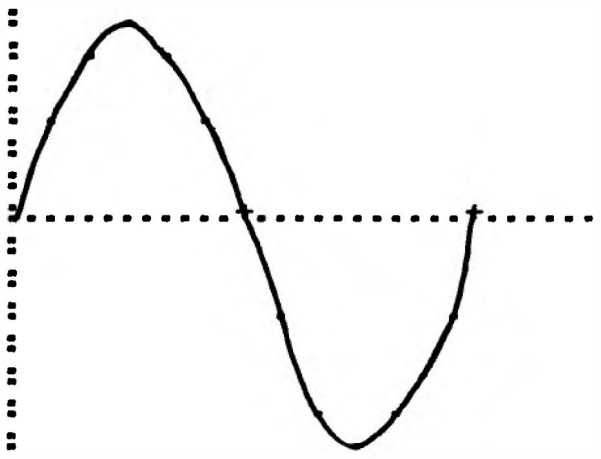
Dealing the Cards

We assume that South is the dealer and the values in the shuffled deck are distributed to the hands in the following order: West, North, East, and South. Remember X(N) was the shuffled array. We used two FOR NEXT LOOPS in statements 180 and 190. J represents the number of cards dealt in each hand and K represents the position to which a card is being dealt. The FOR - NEXT loops make this part easy. We are assuming there are 4 players which is standard for most card games. Statement 235 prints the cards as they are being dealt.

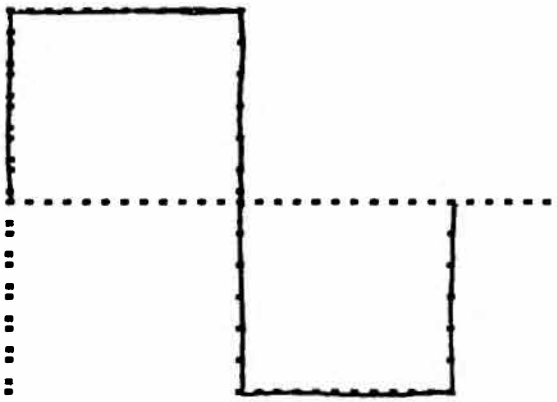
DATA Statements

The data is contained in statements 255 - 285. We used a *,<,&, and @ symbol to represent the suits. At first we used the up arrow but our word processor would not accept this so we changed to the @ symbol. You may want to use the up arrow as this looks similar to the symbol for spades.

COMPUTER
SOUNDS (Part 3)



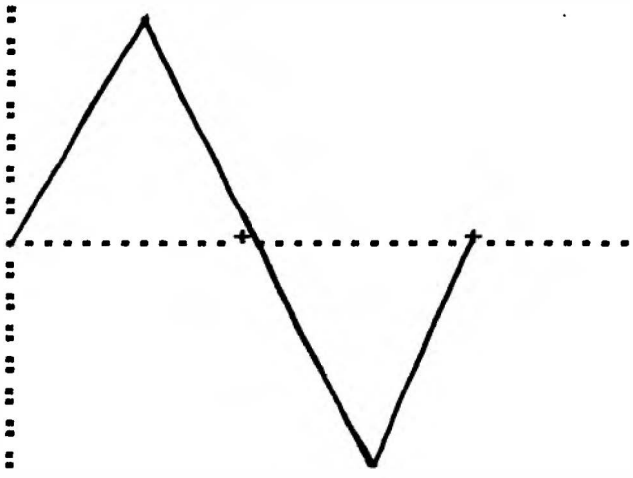
SINE WAVE



SQUARE WAVE

Last month we showed how to generate sounds by writing directly to the Digital to Analog Converter. In the first part of this series we discussed the Sine Wave and last month month we discussed Square Waves. We had a demonstration program that allowed various sounds to be made from the keyboard. Because basic is too slow we used a machine language subroutine that we called from basic to increase the speed. This technique is very useful because basic is convenient for organizing a program and the machine language subroutines allow things to be done faster. This also minimizes the amount of machine language programming that must be done.

This month we want to look at 3 popular waveforms and listen to the sounds that they produce. The basic waveform for all music, voice, and audio waveforms is the sinewave. The triangle wave is composed of a sinewave at the frequency of the triangle plus sinewaves that are odd multiples of the basic frequency. These higher frequencies are called harmonics. Square waves also have these odd harmonic frequencies but they are larger in amplitude than the triangle wave. The triangle and square waves are easy to generate and this is the reason we are discussing them. A square wave just alternates from the maximum level to the minimum level. A triangle wave is generated by drawing straight lines or increasing or decreasing its value by one unit as the time increases.



TRIANGLE WAVE

We used an approach for our

demonstration program that is similar to the one we used last month. To obtain the desired speed we wrote a machine language subroutine and were able to carry it in the first remark statement. Remember you can do this if there are no zeros in the remark statement. You can't save the program in ASCII format because some of the machine code values may be higher than 128 and ASCII will not handle these values. Also you may have trouble editing your program but you will not have any trouble saving the program as a Basic program after it is entered correctly. We would recomment that you enter statement 10 with a remark statement and then about 35 spaces which you can fill in with the data last. Let's look at the machine language subroutine.

Machine
Language Subroutine

We reserved memory locations 500 - 535 to store the values of the waveforms. Since we are comparing 3 different waveforms, we will store the values for each waveform in these locations. Basic will do this for us and the machine language subroutine will take the values from memory and store them in the PIA which controls the digital to analog converter. Location 499 is the delay for each location and location 498 is the duration of the tones or the number of times it is repeated. A listing and discussion of the subroutine follows:

```
7687 LDX I  500 ' Load the X
     register with the value
     500.
7690 LDA X R+  ' Load A with
     the memory X is pointing
     to and increment X.
7692 STA E  65312 'Store A in
     65312 which is the D/A
     Converter
7695 LDB E  499 ' Load B with
     the value in 499
7698 DECB ' B=B-1
7699 BNE  7698 ' Go to 7698 if
     B is not zero
7701 CMPXI  535 ' Compare X
     with the value 535
7704 BLS  7690 ' Branch to 7690
     if X is less than or the
     same as 1535
7706 RTS  'Return from Sub.

7707 LDA E 498 'Load A with the
     value in memory 498
7710 STA E 497 'Store A in 497
7713 BSR  7687 'Branch to the
     subroutine at 7687
7715 DEC E  497 'Decrease the
     value in 497 by 1
7718 BNE  7713 'Branch if not
     equal to 7713
7720 RTS  'Return from sub.
```

The subroutine called from basic by the EXEC command is at 7707 if the Basic program starts with 30,1 in locations 25 and 26. To force basic to start at this location do the following: POKE 25,30: POKE 256 * 30,0 : NEW. This subroutine calls the subroutine at 7687 with the instruction in 7713. It counts the number of times the subroutine at 7687 is called with statements 7715 to 7718. Notice that it is usually easier to write programs with subroutines whether in basic or machine language because this groups each task into a subroutine.

The Subroutine at 7687 takes the values in locations 500 - 535 and presents them to the A/D converter. The time the machine language program stays on each location is determined by the B register which was loaded with location 499. This effectively controls the frequency. All of these parameters are set up from basic before calling the machine language subroutine.

Sound
Program Discussion

The program first calculates the location of the machine lan-

guage subroutine contained in statement 10. This is statement 25. Next a menu is presented that allows one of the 3 waveforms or the time to be selected. You select the option you want by entering the number to the left of the desired feature. If you just press the enter key then the last sound is repeated. The subroutines start in locations 200, 400, 600, and 800. The first 3 subroutines calculate and store values for the waveforms in locations 500 - 535. The subroutine at 800 allows a new time to be entered.

After the program is entered you will want to poke the machine language subroutine into the locations reserved in statement 10. After this is done save the program before you run it. Then run the program. You will be able to compare the 3 waveforms. The Sine waveform is the purest of the 3 and will not have any higher frequency sounds. The square wave has the most harmonics and will sound the fullest. The triangle wave is in between.

```
10 '

12 ?"SOUND LEARNING PROGRAM
14 ?"THIS PGM IS SND-2
16 ?"cOPYRIGHT (c) 1984
18 ?"dYNAMIC eLECTRONICS iNC
19 PRINT
20 PRINT"ML PGM IS IN STMT 10
22 SOUND 100,10:POKE65315,255
25 U=256*PEEK(25)+27
30 PRINT"DATA IS IN LOCATIONS
   500-535
40 PRINT"497=CYCLE COUNTER,
   498=DURATION
50 PRINT"499 IS THE TIME
100 T=PEEK(499):IFT=0 THEN T=1:
    POKE 499,1
105 PRINT"TIME IS "T
110 PRINT"1 SINE WAVE
120 PRINT"2 TRIANGLE WAVE
130 PRINT"3 SQUARE WAVE
140 PRINT"4 CHANGE THE TIME
160 INPUT X
170 IF X=0 THEN 190
180 ON X GO SUB 200,400,600,800
190 PRINT:PRINTW$:EXEC U:
    GO TO 100
200 PRINT"THIS LOADS SINE VALUES
    INTO MEM.
205 P=3.14159:X=P/18
210 FOR J=0 TO 35: A=X*J:
    Y=SIN(A):Z=31+31*Y
220 W=4*Z:POKE J+500,W
225 NEXT J
230 W$="SINE WAVE
235 RETURN
400 PRINT"THIS LOADS A TRIANGLE
    WAVE
405 FOR J=0 TO 17
410 V=14*J
415 POKE 500+J,V: POKE 536-J,V
420 NEXT J
425 POKE518,252
435 W$="TRIANGLE WAVE"
440 RETURN
600 PRINT"THIS LOADS SQUARE
    WAVES
605 FOR J=500 TO 517
610 POKE J,120
612 NEXT J
615 FOR J=518 TO 535
620 POKE J,0
625 NEXT J
645 W$="SQUARE WAVE
650 RETURN
800 INPUT"ENTER NEW TIME";T
810 POKE 499,T:RETURN
```

After you have entered the program then the machine language values need to be poked into statement 10. If the beginning vectors are at 30,1 as stated earlier then the values to poke are as follows in decimal and hex.

Memory		Value	
7687	(1E07)	142	(8E)
7688		1	(01)
7689		244	(F4)
7690	(1E0A)	166	(A6)
7691		128	(80)
7692		183	(B7)
7693	(1E0D)	255	(FF)
7694		32	(20)
7695		246	(F6)
7696	(1E10)	1	(01)
7697		243	(F3)
7698		90	(5A)
7699		38	(26)
7700	(1E14)	253	(FD)

7701		140	(8C)
7702		2	(02)
7703		23	(17)
7704	(1E18)	35	(23)
7705		240	(F0)
7706		57	(39)
7707		182	(B6)
7708		1	(01)
7709		242	(F2)
7710	(1E1E)	183	(B7)
7711		1	(01)
7712		241	(F1)
7713		141	(8D)
7714		228	(E4)
7715		122	(7A)
7716		1	(01)
7717	(1E25)	241	(F1)
7718		38	(26)
7719		249	(F9)
7720	(1E28)	57	(39)

The following keyboard program will allow you to enter the values.

```
FOR J=7687 TO 7720:?J:INPUT X:
POKE J,X: NEXT J
```

After the values have been entered then you can save the program and later reload it and the ML program will be contained within the program.

* * * * * * * * * * * * * * * * * *

OPERATING HINT

When entering programs use a "?" whenever you want something printed instead of typing "PRINT". This can be from the keyboard or within Basic Program Statements. Example: ? MEM or PRINT MEM displays the available memory left. Also when entering remarks in Basic Program Steps use an "'" instead of typing "REM".

* * * * * * * * * * * * * * * * * *

QUESTIONS & ANSWERS

QUESTION: Can a second Color Computer be used as a printer spooler?

ANSWER: Yes this would be a very economical way to obtain a printer spooler. A spooler is a memory that accepts characters quickly from the computer and passes them on to the printer at the printer's slower rate. This quickly frees the computer. With the low cost of Color Computers this is now economically attractive. The second computer would have to have software to handle this.

* * * * * * * * * * * * * * * * * *

CLASSIFIED ADS

As we start our second year we will be taking classified and display advertisements. If you have something to sell you can advertise here.

CLASSIFIED AD RATES

1. 10 cents a word, $3 minimum.
2. Name, Address, & Telephone listed free.
3. Send payment with ad.
4. Closing date 10th of the preceeding month. Ex. Nov ad closing is Oct. 10.

* * * * * * * * * * * * * * * * * *

DISPLAY ADS

1. Closing 10th of preceeding month.
2. Rates as follows:

Pages	1 time	3 times	12 times
1	$25	$22	$20
1/2	18	15	13
1/4	10	8	7

* * * * * * * * * * * * * * * * * *

NEXT MONTH

We will be continuing with our discussion on writing programs for large memories plus we will begin a discussion on Computer Video and Graphics.

Please sign me up for one year for the DYNAMIC COLOR NEWS SERVICE. I Want to receive instruction on programming, Computer Theory, Operating Techniques, Computer Expansion plus the Individual Reply to my Computer problems for a special of $10 each. Also I understand that there will be no charge for letters printed with answers in the Newsletter. Cost $15 USA & Canada, $30 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