



*The seventh in a series of tutorials
for the beginner to intermediate machine
language programmer*

Machine Language Made BASIC: Part VII: Around in Circles

By William P. Nee

The circle routine in ROM is certainly one of the most difficult to understand and use. The routine keeps alternating between its BASIC check for syntax, format, etc., and short subroutines required as set-up for the main routine. Since the BASIC checks would interfere with a machine language program, we must bypass them and implement the subroutine on our own. The following locations are used in the circle routine:

Location

- \$B5 color
- ** \$CB/CC horizontal center of circle (X)
- ** \$CD/CE vertical center of circle (Y)
- ** \$CF/D0 radius
- ** \$D1/D2 height to width ratio
- ** \$D3/D4 maximum horizontal coordinate
- ** \$D5/D6 maximum vertical coordinate
- \$D8 circle/line flag 1:
= circle, 0 = line
- \$D9/DA arc end

The locations marked with two asterisks (**) must be scaled for locations, or dimensions, in any PMODE other than

PMODE 4. The CIRCLE routine may be executed from three different addresses, depending on how much of the CIRCLE command you use.

The first step is to load the maximum coordinates into \$D3/D4 and \$D5/D6 by using the subroutine at Address \$9522. This routine not only loads the correct numbers according to the PMODE, it also scales them. Next, enter the coordinates of the circle's center. This is usually done by loading Register D with the X,Y locations (i.e., #\$8060 would be the center of the screen), then storing Register A in \$0CC and Register B in \$CE. We must show Location \$CC as \$0CC because EDTASM+ requires a zero in front of any number that could also be a register, such as A, B, CC, or D.

These locations are scaled by loading Stack U with #\$CB (start of X location) followed by JSR \$9320. Next, load Register A (LDA) with the radius, and store it in Location \$D0. Scale it by loading Stack U with #\$CF followed by JSR \$9320. Load Register B with the desired color byte and store it in \$B5.

The height-to-width subroutine at \$9EC9 converts the desired height/width ratio to a two-byte number and stores it in Location \$D1/D2. The following chart identifies the number to be placed in Location \$D1/D2 for each height-to-width ratio and possible PMODE screen. The numbers for PMODE 3 and 2 are twice those of PMODE 4, 1, and 0.

Height-to-Width Ratio	PMODE 4, 1 or 0	PMODE 3 or 2
.1	#\$0019	#\$0032
.2	#\$0033	#\$0066
.3	#\$004C	#\$0098
.4	#\$0066	#\$00CC
.5	#\$0080	#\$0100
.6	#\$0099	#\$0132
.7	#\$00B3	#\$0166
.8	#\$00CC	#\$0198
.9	#\$00EC	#\$01D8
1.0	#\$0100	#\$0200
1.5	#\$0180	#\$0300
2.0	#\$0200	#\$0400
2.5	#\$0280	#\$0500

If you want your program to run in any PMODE, include the subroutine, which checks the PMODE and doubles the height-to-width ratio when necessary:

```

LDX  #$**  ** use two-byte
                number from chart
LDA  $B6    get PMODE
BITA #2     register A AND 2
BEQ  NEXT  PMODE 4, 1,
                or 0 will equal 0
TFR  X,D   height-to-width
                ratio to Register D
LEAX D,X   double it
NEXT STX  $D1
    
```

Store the number 1 in Location \$D8 to indicate that no lines should be drawn. Try clearing this location in a program and see what happens.

The arc subroutine at \$9FED converts an arc decimal to a two-byte number stored in Location \$D9/DA.

Bill Nee bucked the "snowbird" trend by retiring to Wisconsin from a banking career in Florida. He spends the long, cold winters writing programs for his CoCo.

Many of these arcs and their two-byte equivalents are in the following chart:

Arc in Degrees	Arc in Degrees
0 = #\$0600	210 = #\$0205
30 = #\$0605	225 = #\$0300
45 = #\$0700	240 = #\$0302
60 = #\$0702	270 = #\$0400
90 = #\$0000	300 = #\$0405
120 = #\$0005	315 = #\$0500
135 = #\$0100	330 = #\$0502
150 = #\$0102	360 = #\$0600
180 = #\$0200	

Load Register D with the desired arc end two-byte equivalent, and store it in Location \$D9/DA. Then load Register D with the arc-start two-byte equivalent and JSR \$9EFD. Be sure to save whatever is in registers A or B prior to executing this routine if you will need them again.

Listing 1 begins by drawing a circle in the center of the screen. It continues drawing circles — each two steps larger than the previous one — until the radius reaches 90. Instead of using the standard start-up assigned by the computer, COLOR 3,0, this program uses COLOR 0,3, which sets a buff background and draws black circles.

If the height-to-width ratio is 1 and the routine draws a full circle with no arcs, you can stop after scaling the radius and JSR \$9EC2. Again, be sure

JSR \$9EC2	JSR \$9EDF	JSR \$9EFD
SET MAX X,Y	PLUS -	PLUS -
SET CENTER/SCALE	SET COLOR	STORE "1" IN \$D8
SET RADIUS/SCALE	SET H/W RATIO	SET ARC END
		SET ARC START

Figure 1: Routines for Drawing Circles or Arcs

that you have already saved registers A and B. This routine also checks Location \$C2 for a PSET or PRESET, stores a '1' in Location \$D8 (which indicates that no lines will be drawn), and assigns the foreground color to Location \$B5.

If you want to specify a height-to-width ratio other than 1, load that ratio's two-byte equivalent into Register X, the desired color into \$B5, and then load JSR \$9EDF. Using this address, the height-to-width ratio will automatically be scaled (doubled) for PMODE 3 or 2. The routine also checks Location \$C2 for a PSET or PRESET and stores a '1' in Location \$D8.

Assigning arc starts and arc ends will require that you use the height-to-width ratio chart described earlier in this article. Store the correct two-byte height-to-width ratio (according to the PMODE) in \$D1/D2; store the two-byte arc end in \$D9/DA; load Register D with the two-byte arc start, then with

JSR \$9EFD.

More than one arc can be drawn by repeating the arc portion of the program as often as desired. Color changes can be made by putting the desired color into Location \$B5 prior to executing the routine. Listing 2 is a program that draws different colored arcs of 60 degrees at increasing radii.

Since the first program is in PMODE 4, I didn't need to scale any of the numbers, but I did so to show how it is done. It is good practice to always scale since you may want to change the PMODE later.

The routines for drawing circles or arcs are found in Figure 1.

Before assembling either program, set Location \$FF/100 to #\$2000 and execute GC006. If you run the assembled programs from BASIC, clear enough memory by entering CLEAR 200, &H3000-1, and change the SWI in Line 670 to RTS. □

Listing 1: CIRCLES

		00100	*\$FF/100=#\$2000		
3000		00110	ORG	\$3000	
3000 C6	04	00120	START LDB	#4	PMODE 4
3002 BD	9628	00130	JSR	\$9628	
3005 C6	01	00140	LDB	#1	PAGE 1
3007 BD	9653	00150	JSR	\$9653	
300A 0F	B2	00160	CLR	\$B2	CLEAR FOREGROUND
300C C6	03	00170	LDB	#3	
300E D7	B3	00180	STB	\$B3	BACKGROUND COLOR
3010 BD	9542	00190	JSR	\$9542	COLOR 0,3
3013 C6	01	00200	LDB	#1	GRAPHICS SCREEN
3015 BD	95AA	00210	JSR	\$95AA	
3018 C6	01	00220	LDB	#1	COLOR SET 1
301A BD	9682	00230	JSR	\$9682	
301D BD	9522	00240	JSR	\$9522	SET MAXIMUM X,Y AND SCALE THEM
3020 CC	8060	00250	LDD	#\$8060	CENTER OF THE CIRCLE(128,96)
3023 97	CC	00260	STA	\$0CC	X LOCATION
3025 D7	CE	00270	STB	\$CE	Y LOCATION
3027 CE	00CB	00280	LDU	#\$CB	START OF X LOCATION
302A BD	9320	00290	JSR	\$9320	SCALE X,Y
302D C6	02	00300	LDB	#2	SIZE OF FIRST RADIUS
302F 34	04	00310	LOOP PSHS	B	SAVE RADIUS

3031	D7	D0	00320	STB	\$D0	RADIUS LOCATION
3033	CE	00CF	00330	LDU	#\$CF	START OF RADIUS LOCATION
3036	BD	9320	00340	JSR	\$9320	SCALE RADIUS
3039	BD	9EC2	00350 CIRCLE	JSR	\$9EC2	CIRCLE(X,Y),R
303C	35	04	00360	PULS	B	GET THE RADIUS
303E	CB	02	00370	ADDB	#2	INCREASE IT
3040	C1	5A	00380	CMPB	#90	MAXIMUM RADIUS YET?
3042	23	EB	00390	BLS	LOOP	
3044	BD	ADFB	00400 DONE	JSR	\$ADFB	WAIT FOR INPUT
3047	5F		00410	CLRB		TEXT SCREEN
3048	BD	95AA	00420	JSR	\$95AA	
304B	3F		00430	SWI		RTS IF IN BASIC
		3000	00440	END	START	

000000 TOTAL ERRORS

Listing 2: ARCS

```

00100 * $FF/100=#$2000
3000
3000 C6 03 00110 ORG $3000
3002 BD 9628 00120 START LDB #3 PMODE 3
3005 C6 01 00130 JSR $9628
3007 BD 9653 00140 LDB #1 PAGE 1
300A BD 9542 00150 JSR $9653
300D C6 01 00160 JSR $9542 PCLS
300F BD 95AA 00170 LDB #1 GRAPHICS SCREEN
3012 C6 01 00180 JSR $95AA
3014 BD 9682 00190 LDB #1 COLOR SET 1
3017 BD 9522 00200 JSR $9682
301A CC 8060 00210 JSR $9522 SET MAXIMUM X,Y AND SCALE THEM
301D 97 CC 00220 LDD #$8060 CENTER OF CIRCLE(128,96)
301F D7 CE 00230 STA $0CC X LOCATION
3021 CE 00CB 00240 STB $CE Y LOCATION
3024 BD 9320 00250 LDU #0CB START OF X LOCATION
3027 C6 02 00260 JSR $9320 SCALE X,Y
3029 34 04 00270 LDB #2 SIZE OF FIRST RADIUS
302B D7 D0 00280 LOOP PSHS B SAVE THE RADIUS
302D CE 00CF 00290 STB $D0 RADIUS LOCATION
3030 BD 9320 00300 LDU #0CF START OF RADIUS LOCATION
3033 8E 0100 00310 JSR $9320 SCALE THE RADIUS
3036 96 B6 00320 LDX #0100 HEIGHT TO WIDTH RATIO = 1
3038 85 02 00330 LDA $B6 PMODE
303A 27 04 00340 BITA #2 PMODE 4,1, OR 0 WILL = 0
303C 1F 10 00350 BEQ NEXT
303E 30 8B 00360 TFR X,D REGISTER X TO REGISTER D
3040 9F D1 00370 LEAX D,X ADD REGISTER D TO REGISTER X
3042 CC 0155 00380 NEXT STX $D1 HEIGHT TO WIDTH RATIO LOCATION
3045 97 D8 00390 LDD #0155
3047 D7 B5 00400 STA $D8 NO LINES, JUST CIRCLES
3049 CC 0702 00410 STB $B5 COLOR
304C DD D9 00420 LDD #0702 END OF ARC 1 - 60 DEGREES
304E CC 0600 00430 STD $D9 END OF ARC LOCATION
00440 LDD #0600 START OF ARC 1 - 0 DEGREES

```

3051	BD	9EFD	00450	JSR	\$9EFD	DRAW THE ARC
3054	CC	01AA	00460	LDD	##01AA	
3057	97	D8	00470	STA	\$D8	NO LINES
3059	D7	B5	00480	STB	\$B5	COLOR
305B	CC	0200	00490	LDD	##0200	END OF ARC 2 - 180 DEGREES
305E	DD	D9	00500	STD	\$D9	END OF ARC LOCATION
3060	CC	0005	00510	LDD	##0005	START OF ARC 2 - 120 DEGREES
3063	BD	9EFD	00520	JSR	\$9EFD	DRAW THE ARC
3066	CC	01FF	00530	LDD	##01FF	
3069	97	D8	00540	STA	\$D8	NO LINES
306B	D7	B5	00550	STB	\$B5	COLOR
306D	CC	0405	00560	LDD	##0405	END OF ARC 3 - 300 DEGREES
3070	DD	D9	00570	STD	\$D9	START OF ARC LOCATION
3072	CC	0302	00580	LDD	##0302	START OF ARC 3 - 240 DEGREES
3075	BD	9EFD	00590	JSR	\$9EFD	DRAW THE ARC
3078	35	04	00600	PULS	B	GET THE RADIUS
307A	CB	02	00610	ADDB	#2	INCREASE IT
307C	C1	5A	00620	CMPB	#90	MAXIMUM RADIUS YET?
307E	23	A9	00630	BLS	LOOP	
3080	BD	ADFB	00640	JSR	\$ADFB	WAIT FOR INPUT
3083	5F		00650	CLRB		TEXT SCREEN
3084	BD	95AA	00660	JSR	\$95AA	
3087	3F		00670	SWI		RTS IF IN BASIC
		3000	00680	END	START	

00000 TOTAL ERRORS

HOLIDAY SPECIALS from SPORTSware

Offer good thru 31 Jan 89. Previous orders excluded.

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

STAND ALONE WGD SCENARIOS

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

NEW YEAR'S SALE

All Software Listed Below

\$15.00 each until Jan. 31, 1989

PROGRAM TITLE	GRADES	MEMORY	PRICE
---------------	--------	--------	-------

LANGUAGE ARTS

Beyond Words 1-3 parts	3-5	32K-Ext.	19.95
Beyond Words 2-3 parts	6-8	32K-Ext.	19.95
Beyond Words 3-3 parts	9-12	32K-Ext.	19.95
Vocabulary 1-1000 words	3-5	32K-Ext.	19.95
Vocabulary 2-1000 words	6-8	32K-Ext.	19.95
Vocabulary 3-1000 words	9-12	32K-Ext.	19.95
Context Clues	4,5,6, or 7	16K-Ext.	17.95

Reading Aids - 4 parts	2-4	16K-Ext.	19.95
King Author - writing tool	2-6	16/32 Ext.	29.95
Cocowheel of Fortune	4-up	32K-	19.95

CRITICAL THINKING PROBLEMS

Factory by Sunburst	4-up	32K-disk	44.95
Pond by Sunburst	2-up	32K-disk	44.95
Teasers by Tobbs-Sunb	4-up	32K-disk	44.95

Computer Island

(718) 948-2748

227 Hampton Green,
Staten Island, N.Y. 10312

*Please add \$1.00 per order for postage. N.Y. residents, please add proper tax