

CLEVELAND AREA TI-994/A USER GROUPS NEWSLETTER

JULY, AUGUST, 1989

OFFICERS	NORTHCOAST	TI-CHIPS	MEETING DATES
PRESIDENT	ERNIE MALNAR 289-7742	MATT ANDEL 676-9759	NORTHCOAST 1:30 P.M. TI-CHIPS 10 A.M.
V. PRESIDENT	MARTY SMOLEY 1-257-1661	GLENN BERNASEK 238-6335	EUCLIDIAN ROOM N.ROYALTON LIBRARY
TREASURER	FRANK JENKINS 283-8526	LIN SHAW 235-3912	EUCLID SQUARE MALL STATE RD & RT 82
MEMBERSHIP	CHUCK POULIN 731-6475 361 E. 280TH ST. EUCLID, OH 44132	JOHN PARKEN 331-2830 4172 W. 217TH ST. FAIRVIEW PARK, OH 44126	THIRD SATURDAY THIRD SATURDAY
SECRETARY	CHUCK POULIN 731-6475	MARY PHILLIPS 582-5009	JULY 15, 1989
LIBRARY(DISK)	MARTIN SMOLEY 1-257-1661	HARRY HOFFMAN 631-2354	AUGUST 12, 1989
(TAPE & MODS)	TOM NELLIS 475-4067	JOHN PARKEN 331-2830	SEPTEMBER 16, 1989
(HARD COPY)	DICK ALDEN 1-352-9172		OCTOBER 21, 1989
			NOVEMBER 18, 1989

Note this is our double 'Summer Issue'. We won't be printing again until September. Your staff welcomes this break. We have had to borrow from other newsletters very little this past year and our authors are producing some of the best tips and articles on the TI in the TI world.

When Marty Smoley dropped off his TI-BASE articles, I reminded him he had been at his TI-BASE tutorials for a full year. Our regulars have been Marty, Glenn Bernasek, Paul Newmeyer, and sometimes myself. Our newest contributor, Wes Richardson, has been here only a few months and already has enough material generated for a 'CLEVELAND!' disk. Also, one of Wes' little programs was picked up in the May 'MICROPENDIUM'.

Quoting from the May/June issue of 'ONLINE TODAY', a Comuserve publication, 'With the 1980s coming to an end, experts say that symbols of the decade, including Swatch watches, Pee-wee Herman dolls and 'Baby on Board' signs, will soon become valuable. Likewise, certain models of the personal computer, born in 1977, may someday go for astronomical prices at auctions. Primitive pre-1980 personal computers from companies such as IMSAI, Cromemco, North Star and MITS may someday be worth a mint. Discontinued computers from the early '80s, including Osborne, Commodore PET, early Kaypros, TI-994/A and Adam should also have value...Ironically, as a computer becomes obsolete and its used-market value drops, it becomes precious on the collectable market.' So hang in there, we may have the last laugh yet.

The following is a quick tip from the May issue of PUNN from Portland, OR: The following program will provide a hard copy listing in a 28-character format. If you are using

XBASIC, save it as a merged program. Then with the printer 'on', merge and run this program with the program you want to list. After running, delete lines 2 through 6 and enter LIST 'PIO'.

If you are using Basic, type in lines 2 through 6 after you load the program you want to print. RUN this program, then delete lines 2 through 6 and enter LIST 'PIO'.

```

2 OPEN "PIO"
3 OPEN #2:CHR$(27)CHR$(77)&CHR$(N)
4 OPEN #2:CHR$(27)CHR$(81)&CHR$(N+28)
5 CLOSE #2
6 STOP

```

(If you have SUPER EXTENDED BASIC, to LIST a program in 28 columns, enter LIST 'PIO':28:)

: CONTENTS	:
: EXECUTIVE NOTES - NORTHCOAST.....	2:
: EXECUTIVE NOTES - TI-CHIPS.....	2:
: TI KEY/CHARACTER CODES - Jim Swedlow.....	3:
: TI-BASE TUTORIAL - MARTY SMOLEY - NORTHCOAST.....	4:
: INSTANT CARDS - DEANNA SHERIDAN - NORTHCOAST.....	6:
: GIANT ARTIST POSTERS (GAP) - DEANNA SHERIDAN - NC.....	7:
: JIFFY CARD - DEANNA SHERIDAN - NORTHCOAST.....	7:
: PAGE PRO 99 - DEANNA SHERIDAN - NORTHCOAST.....	8:
: PAGE PRO EXAMPLE - HARRY HOFFMAN - CHIPS.....	9:
: SPINNER - WES RICHARDSON - NORTHCOAST.....	10:
: KEBOARD FIX - GLENN BERNASEK - TI-CHIPS.....	18:

This was a small meeting. About twenty people showed up. Not bad considering it did not rain Saturday and most people had a foot of grass to mow. This is the time of year when everyone has yard work to do and would rather be outside on the weekends if possible. The members who did show up got a great demonstration of TI-LOGO by Wes Richardson. During the meeting many of us admitted to owning Logo but not having much knowledge of how to make it work or what to do with it after that. Wes demonstrated Logo's great graphic capability and the ease with which Logo can perform complicated movement, color and sound operations.

CHEEP DISKS

Cheep disks are back. As many of you know, I was selling top quality double sided double density disks at several meetings. They came with sleeves and write protect tabs in packages of 25 for \$7.50. Of course they went like hotcakes. Well, I will have more available at the regular meetings for the same price, starting in July. This time around there will not be a signup sheet, just bring your money and come-on down to the meetings. Note: Even if you don't need disks, come-on down to the meetings anyway.

MEMBERSHIP RENEWALS

At this time we have a fairly large number of members who are up for renewal. I'd like to ask you all to please send in your \$15.00 and stick with us. We have one of the best groups in the country. We have great meetings, with great demonstrations and a world class newsletter. That old TI of yours is capable of handling any job for the home or for a small business, and the software gets better every day (and dirt cheap too). For those NorthCoast members and anyone else who reads this column; stick with the NorthCoast 99'ers or some other TI group. It's a lot more fun when you associate with great people who have the same interest you do. I find the IBM compatible world to be a rather cold, grey place. The TI world seems more like a family room with a fireplace.

THE NEXT NORTHCOAST MEETING

Deanna Sheridan will do the demo at the next meeting. She will be demonstrating the newest releases from Rodger Merritt. I think it will be a general overview of JIFFY CARD, Giant Artist Posters, FORMSHOP, JIFFY FLYER and PICTURE_IT. Even if she only touches a few points on each item, it should be a great graphics demo and the graphics enthusiasts are a very active part of the NorthCoast group so it should be a really great meeting.

TI-Base >>=>> Graphics Note

I will do another set of CFs and a Tutorial to follow up Wes Richardson's article in the June issue. I just couldn't get it put together for this issue, and I also thought that the Date Field information was important due to questions I have been receiving.

Hope to see all of you at the next meeting. Marty

All of the Chips members who attended the Lima Conference in May were enthusiastic about what they had learned and enjoyed. Much of this was covered by several writers in the last newsletter, so won't be repeated here. The only criticism heard was about the timing of the individual seminars. Overlapping sessions prevented some people from getting to everything they wanted to hear. It was certainly encouraging to hear about the enthusiasm still evident for the "orphan".

Harry Hoffman had had time since May to review "Page Pro 99", a new graphic printing program. He liked it very much, and found it easier to use than "Printers Apprentice." Harry provided printouts of what he had produced with the software and a good demonstration at the meeting of how it works.

Harry and Glenn Bernasek and other reminded all members about "fairware." Much work goes into producing these programs. Although they are primarily "labors of love", monetary compensation is greatly appreciated. So, if you are using a fairware program regularly, take the time to send off a check or money order to its author/authors. They will be encouraged to continue creating useful and entertaining software for all to enjoy.

Mark your calendar with the summer meeting dates. See you then.

CLEVELAND
INDIANS



T I KEY/CHARACTER CODES

KEY	ASC	HEX	NAME	KEY	ASC	HEX	NAME	KEY	ASC	HEX	NAME	KEY	ASC	HEX	NAME
	0	>00	EOL marker	@	64	>40		CTRL ,	128	>80	null	FCTN J	192	>C0	>
FCTN 7	1	>01	AID	A	65	>41		CTRL A	129	>81	ELSE	FCTN K	193	>C1	+
FCTN 4	2	>02	CLEAR	B	66	>42		CTRL B	130	>82	!!	FCTN L	194	>C2	-
FCTN 1	3	>03	DELete	C	67	>43		CTRL C	131	>83	!	FCTN M	195	>C3	*
FCTN 2	4	>04	INSert	D	68	>44		CTRL D	132	>84	IF	FCTN N	196	>C4	/
FCTN =	5	>05	QUIT	E	69	>45		CTRL E	133	>85	GO	FCTN O	197	>C5	^
FCTN 8	6	>06	REDO	F	70	>46		CTRL F	134	>86	GOTO	FCTN Y	198	>C6	
FCTN 3	7	>07	ERASE	G	71	>47		CTRL G	135	>87	GOSUB		199	>C7	flag quoted
FCTN 8	8	>08	Left Arrow	H	72	>48		CTRL H	136	>88	RETURN		200	>C8	flag unquoted
FCTN D	9	>09	Right Arrow	I	73	>49		CTRL I	137	>89	DEF		201	>C9	flag line on
FCTN X	10	>0A	Down Arrow	J	74	>4A		CTRL J	138	>8A	DIM		202	>CA	EOF
FCTN E	11	>0B	Up Arrow	K	75	>4B		CTRL K	139	>8B	END		203	>CB	ABS
FCTN 6	12	>0C	PROC'D	L	76	>4C		CTRL L	140	>8C	FOR		204	>CC	ATN
ENTER	13	>0D	ENTER	M	77	>4D		CTRL M	141	>8D	LET		205	>CD	COS
FCTN 5	14	>0E	BEGIN	N	78	>4E		CTRL N	142	>8E	BREAK		206	>CE	EXP
FCTN 9	15	>0F	BACK	O	79	>4F		CTRL O	143	>8F	UNBREAK		207	>CF	INT
	16	>10		P	80	>50		CTRL P	144	>90	TRACE		208	>D0	LOG
	17	>11		Q	81	>51		CTRL Q	145	>91	UNTRACE		209	>D1	SGN
	18	>12		R	82	>52		CTRL R	146	>92	INPUT		210	>D2	SIN
	19	>13		S	83	>53		CTRL S	147	>93	DATA		211	>D3	SQR
	20	>14		T	84	>54		CTRL T	148	>94	RESTORE		212	>D4	TAN
	21	>15		U	85	>55		CTRL U	149	>95	RANDOMIZE		213	>D5	LEN
	22	>16		V	86	>56		CTRL V	150	>96	NEXT		214	>D6	CHR#
	23	>17		W	87	>57		CTRL W	151	>97	READ		215	>D7	RND
	24	>18		X	88	>58		CTRL X	152	>98	STOP		216	>D8	SEG#
	25	>19		Y	89	>59		CTRL Y	153	>99	DELETE		217	>D9	POS
	26	>1A		Z	90	>5A		CTRL Z	154	>9A	REM		218	>DA	VAL
	27	>1B	ESC	[91	>5B		CTRL .	155	>9B	ON		219	>DB	STR#
	28	>1C		\	92	>5C		CTRL	156	>9C	PRINT		220	>DC	ASC
	29	>1D		^	93	>5D		CTRL =	157	>9D	CALL		221	>DD	PI
	30	>1E	Cursor	~	94	>5E		CTRL @	158	>9E	OPTION		222	>DE	REC
	31	>1F	Edge	¯	95	>5F		CTRL 9	159	>9F	OPEN		223	>DF	MAX
Space	32	>20	Space		96	>60			160	>A0	CLOSE		224	>E0	MIN
!	33	>21		a	97	>61			161	>A1	SUB		225	>E1	RPT#
"	34	>22		b	98	>62			162	>A2	DISPLAY		226	>E2	
#	35	>23		c	99	>63			163	>A3	IMAGE		227	>E3	
\$	36	>24		d	100	>64			164	>A4	ACCEPT		228	>E4	
%	37	>25		e	101	>65			165	>A5	ERROR		229	>E5	
&	38	>26		f	102	>66			166	>A6	WARNING		230	>E6	
'	39	>27		g	103	>67			167	>A7	SUBEXIT		231	>E7	
(40	>28		h	104	>68			168	>A8	SUBEND		232	>E8	NUMERIC
)	41	>29		i	105	>69			169	>A9	RUN		233	>E9	DIGIT
*	42	>2A		j	106	>6A			170	>AA	LINPUT		234	>EA	UALPHA
+	43	>2B		k	107	>6B			171	>AB			235	>EB	SIZE
,	44	>2C		l	108	>6C			172	>AC			236	>EC	ALL
-	45	>2D		m	109	>6D			173	>AD			237	>ED	USING
.	46	>2E		n	110	>6E			174	>AE			238	>EE	BEEP
/	47	>2F		o	111	>6F			175	>AF			239	>EF	ERASE
0	48	>30		p	112	>70		CTRL O	176	>B0	THEN		240	>F0	AT
1	49	>31		q	113	>71		CTRL I	177	>B1	TO		241	>F1	BASE
2	50	>32		r	114	>72		CTRL 2	178	>B2	STEP		242	>F2	
3	51	>33		s	115	>73		CTRL 3	179	>B3	,		243	>F3	VARIABLE
4	52	>34		t	116	>74		CTRL 4	180	>B4	;		244	>F4	RELATIVE
5	53	>35		u	117	>75		CTRL 5	181	>B5	:		245	>F5	INTERNAL
6	54	>36		v	118	>76		CTRL 6	182	>B6)		246	>F6	SEQUENTIAL
7	55	>37		w	119	>77		CTRL 7	183	>B7	(247	>F7	OUTPUT
8	56	>38		x	120	>78		FCTN ,	184	>B8	&		248	>F8	UPDATE
9	57	>39		y	121	>79		FCTN .	185	>B9			249	>F9	APPEND
:	58	>3A		z	122	>7A		FCTN /	186	>BA	OR		250	>FA	FIXED
;	59	>3B		[123	>7B		CTRL /	187	>BB	AND		251	>FB	PERMANENT
<	60	>3C			124	>7C			188	>BC	XOR		252	>FC	TAB
=	61	>3D		}	125	>7D		FCTN ;	189	>BD	NOT		253	>FD	#
>	62	>3E		~	126	>7E		FCTN B	190	>BE	=		254	>FE	VALIDATE
?	63	>3F		FCTN V	127	>7F	del	FCTN H	191	>BF	<		255	>FF	EOF marker

**TI-BASE - From INSCEBOT
TUTORIAL 10.1.1 By Martin Smoley
NorthCoast 99'ers - June 17, 1989
Copyright 1989 By Martin A. Smoley**

I am reserving the copyright on this material, but I will allow the copying of this material by anyone under the following conditions. (1) It must be copied in its entirety with no changes. (2) If it is retyped, credit must be given to myself and the NorthCoast 99ers, as above. (3) The last major condition is that there may not be any profit directly involved in the copying or transfer of this material. In other words, Clubs can use it in their newsletters and you can give a copy to your friend as long as its free.

This tutorial (I hope) will give you some ideas on how to use the DATE or, D-type fields. Not too long ago I was under the impression that you couldn't do much with a Date type field. Of course I knew you could sort a date field, and find records through the date field, but I didn't think past that point. Then I got into a situation where I needed to calculate the age of a ewe. That's right, I said a ewe, the mother of a lamb. I couldn't do a thing with D fields using MM, DD or YY, so I called Dennis for some help, and did I ever feel like a dummy. Dennis said you can extract a month by using MONTH, a day by using DAY and a year by using YEAR. "Ask the right question and you'll get the right answer (I thought)." Not from Dennis, from TI-Base. Now I'll try to explain it to you. NOTE: In the CFs included in this article the lines with numbers are actual CF or program lines and the lines starting with REC or 0000 are the product of the DISPLAY commands that immediately preceded them. If you are entering these CFs, enter only the lines that have line numbers, but do not enter the line numbers. The (nnn) means refer to a specific line number.

A date field is used to store a date in the form "MM/DD/YY". "I'm sure that this is old news for most of you." This form can be used in a Db field, or in a local variable. An example would be LOCAL BORN D 8 (005). TIB then creates a local variable space 8 units in length under the name BORN and the designation D Type. D type enables TIB to perform Date operations on whatever data it finds stored in that field or variable. BORN is presently empty. REPLACE BORN WITH "02/12/43" (012) would store the date February 12, 1943 in BORN in it's proper form (MM/DD/YY). From this point there are many things that TIB can do with the data stored in BORN. If you enter LOCAL MO N 3 (007) and then REPLACE MO WITH MONTH(BORN) (015), TIB would extract 02 from BORN and place a copy of it in MO. The command DAY(BORN) would extract 12 and YEAR(BORN) would extract 43. If you created AGE N 3 and the current date "06/14/89" was in a date field named CURDT, REPLACE AGE WITH YEAR(CURDT) - YEAR(BORN) (020) would place 46 in AGE. Unfortunately it doesn't work in the other direction. REPLACE YEAR(BORN) WITH "45" does not work (as far as I can tell). If you enter REPLACE BORN WITH "45" (023), the 45 will be place in the far left portion of BORN, which is the month area. DISPLAY BORN would then produce (45), without the parenthesis. Concatenation (|) can be used to get the results you want as far as placing data into a date field (025). You should notice that the second "/" was eliminated (024) to allow for the right most space in AGE (004) which has a length of 3.

```
001 CLEAR
002 CLOSE ALL
003 CLEAR LOCAL
004 LOCAL AGE N 3
005 LOCAL BORN D 8
006 LOCAL CURDT D 8
007 LOCAL MO N 3
008 LOCAL DY N 3
009 LOCAL YR N 3
010 LOCAL TEST N 6
011 DISPLAY BORN,CURDT,AGE
```

```
-----
REC   BORN      CURDT      AGE
0000
```

```
012 REPLACE BORN WITH "02/12/43"
013 REPLACE CURDT WITH "06/18/89"
014 DISPLAY BORN,CURDT,AGE
```

```
-----
REC   BORN      CURDT      AGE
0000  02/12/43  06/18/89
```

```
015 REPLACE MO WITH MONTH(BORN)
016 REPLACE DY WITH DAY(BORN)
017 REPLACE YR WITH YEAR(BORN)
018 DISPLAY MO,DY,YR,AGE
```

```
-----
REC   MO   DY   YR   AGE
0000   2   12  43
```

```
019 REPLACE AGE WITH YEAR(CURDT);
020      - YEAR(BORN)
021 DISPLAY BORN,CURDT,AGE,TEST
```

```
-----
REC   BORN      CURDT      AGE  TEST
0000  02/12/43  06/18/89   46
```

```
022 REPLACE BORN WITH "45"
023 DISPLAY BORN,CURDT,AGE,TEST
```

```
-----
REC   BORN      CURDT      AGE  TEST
0000   45          06/18/89   46
```

```
024 REPLACE CURDT WITH "00/00" | AGE
025 DISPLAY BORN,CURDT,AGE,TEST
```

```
-----
REC   BORN      CURDT      AGE  TEST
0000   45          00/00  46   46
```

```
026 REPLACE BORN WITH "06/31/44"
027 REPLACE TEST WITH DAY(BORN);
      * MONTH(BORN)
028 DISPLAY BORN,CURDT,AGE,TEST
```

```
-----
REC   BORN      CURDT      AGE  TEST
0000  06/31/44  00/00  46   46   18
```

```
029 RETURN Copyright Martin A. Smoley
030 *                               1989
031 *                               TDT3/C
```

Continued Next Page.

**TI-BASE - From INSCEBOT
TUTORIAL 10.1.2 By Martin Smoley
NorthCoast 99'ers - June 17, 1989
Copyright 1989 By Martin A. Smoley**

After you become familiar with their basic functions you can do some interesting things with Date type fields, but you must be careful to remember what you have put in a field and what its position is. For example, REPLACE DATE WITH "Ma/rt/in" will place the character string Ma/rt/in into DATE. You could store Ma/rt/in in a DB and retrieve it later. However, TIB will recognize these as Characters and will not allow you to do anything with them. If you could multiply Ma by 2.2, I can't imagine what the result might be. But on the other hand, TIB seems to be able to recognize numbers which have been entered as characters and then place them into a date type field. I have attempted to show the different aspects of this theory in TDT5 and TDT6. Notice that DATEC is a character variable (006). I then placed "12345.78" into DATEC as a character string (012). The reason I did it in this manner was to make sure that TIB considered 12345.78 to be characters at this point, even though it looks like a number. I then transferred DATEC to DATE (013). DATE is a D or Date type field (005). I was then able to do any Date type function involving DATE, including multiplication of its parts (019), or multiplication by a constant, or LITERAL, (020). Then I went through the same steps using a numeric (N) type field. Notice in line # 24 that TMP (004) is a numeric field and that 77.77 has no quotes around it. I am attempting to guarantee that 77.77 is a number. Then I placed TMP into DATE (026) and performed the Date type functions on 77.77. The reason I added the zeros was to demonstrate what I said previously, that you need to remember the exact position of the data in a field if you expect to use it in this manner. TMP has a width of 8 (004). This means it would fit right into a Date type field. When I added the "00", I moved the spacing to put "7." in the year portion of DATE. Check the " / / " spacing to see where the slashes are located. This means, when YR is multiplied by 100 (032), TIB is multiplying 7 by 100. If you work with this idea, understand it and are very careful, you could use the date type field to extract specific parts of a number. A very simple demonstration of this would be to place a dollar and cents type number into a date field with the cents portion in the year segment of the date field. You could then use REPLACE CENTS WITH YR(DATE) to extract the cents, if that's what you need. Then REPLACE dollars WITH dollars - CENTS would give you the whole dollar amount. I realize that in most cases this idea is too cumbersome to use, but if there is an instance when no other procedure will work, this idea just might do the trick.

SORT by INSCEBOT

I don't know if SORT is the name that will be used, but sort is what it does. I am currently testing this new sort program from those TI-Base guys and it looks great. It will sort TI-Base files, Fix file, Var files and hopefully soon Basic display files. I've tried it on TIB files and it's real fast and easy to use. It should be available soon as a separate disk for under \$15.00 (I think). The fact that it works on many different file types will make it a very useful program.

```
001 CLEAR I have pulled TDT5 and TDT6 together
002 CLOSE ALL to save space. I hope this does not
003 CLEAR LOCAL confuse everyone.
004 LOCAL TMP N 8 2
005 LOCAL DATE D 8
006 LOCAL DATEC C 8
007 LOCAL MO N 3
008 LOCAL DY N 3
009 LOCAL YR N 3
010 LOCAL TEST N 12 2
011 *****
012 REPLACE DATEC WITH "12345.78"
013 REPLACE DATE WITH DATEC
014 DISPLAY DATE, TMP, DATEC
```

```
-----
REC DATE TMP DATEC
0000 12345.78 12345.78
-----
```

```
015 REPLACE MO WITH MONTH(DATE)
016 REPLACE DY WITH DAY(DATE)
017 REPLACE YR WITH YEAR(DATE)
018 DISPLAY DATE, MO, DY, YR, TEST
```

```
-----
REC DATE MO DY YR TEST
0000 12345.78 12 45 78
-----
```

```
019 REPLACE TEST WITH DY * YR
020 REPLACE TMP WITH YR * 2.2
021 DISPLAY DATE, TMP, TEST
```

```
-----
REC DATE TMP TEST
0000 12345.78 171.60 3510.00
-----
```

```
022 * TDT5/C
023 *****
024 REPLACE TMP WITH 77.77
025 DISPLAY DATE, TMP, TEST
```

```
-----
REC DATE TMP TEST
0000 12345.78 77.77 3510.00
-----
```

```
026 REPLACE DATE WITH "00" ; TMP
027 REPLACE MO WITH MONTH(DATE)
028 REPLACE DY WITH DAY(DATE)
029 REPLACE YR WITH YEAR(DATE)
030 DISPLAY DATE, MO, DY, YR, TMP
```

```
-----
REC DATE MO DY YR TMP
0000 00 77. 0 0 7 77.77
>>==>> / / <<==<< Note Spacing
-----
```

```
031 *
032 REPLACE TEST WITH YR * 100
033 DISPLAY DATE, YR, TEST
```

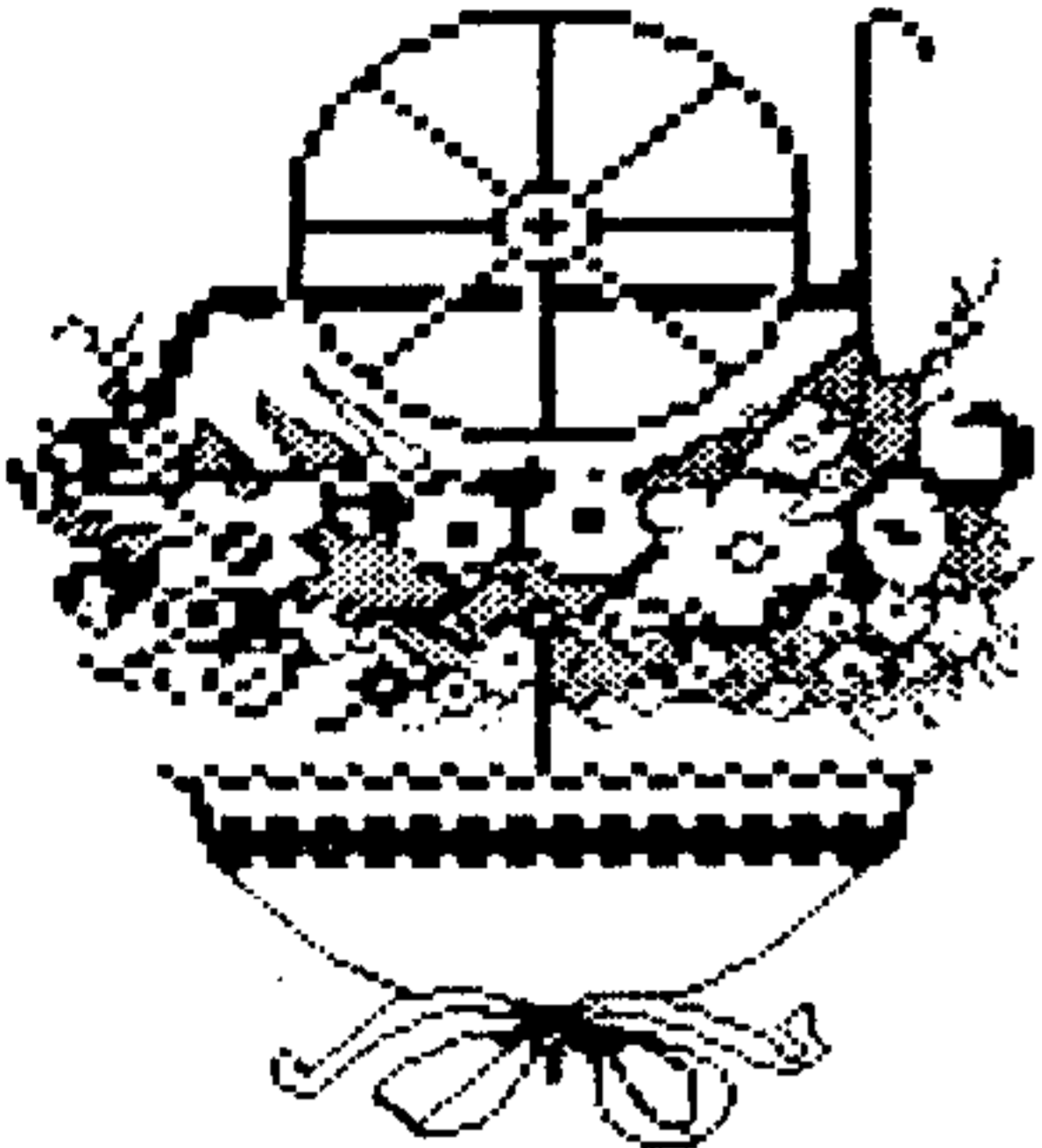
```
-----
REC DATE YR TEST
0000 00 77. 7 700.00
-----
```

```
034 RETURN Copyright Martin A. Smoley
035 * 1989
036 * TDT6/C
```

Continued Next Month.



Thinkin' of you



_____Begin left side message here_____

How would you like to print out your own "Thinkin of You" type note cards and text at the same time.?

This clever idea comes from a file on the new PLUS! disk distributed by Jack Sughrue. Anne Dhein (who was about the first person to show us how to do graphics with TI-Writer) has developed a template to do just that.

As you might suspect, you create your graphic in TI-ARTIST as an instance. Then turn it upside down with the MOVE command before saving. Then use any of several "convert" programs to put it into TI-Writer format. She has provided one on the disk, there is the one by TRIO+ Software, and the one on PICTURE_IT (which also does double density.) Depending on the size

_____Erase guidelines_____

_____Begin right side message here_____

of the instance, this can take some time, so you may only want to do this with a graphic you will use many times.

Check the back of this card just for fun.

I have left in the guidelines so that you can see how the template works.

I have not even looked at the rest of the files on the new PLUS! disk. If this gets you curious, you can order the whole disk from Jack Sughrue, Box 459, East Douglas, MA 01516 for \$10.00

DEANNA SHERIDAN, NORTHCOAST 99ERS
CLEVELAND, OHIO

P.S. Fold this page in quarters to see how it looks as a card!

_____before printing._____

GIANT ARTIST POSTERS (GAP)
Deanna Sheridan NorthCoast 99ers
Cleveland, Ohio

We were fortunate to be one of the first clubs to have on hand another fun program for printing graphics at the Lima Conference.

Giant Artist Posters uses TI Artist instances and prints them from 4 X to 24 X magnification. You have two choices when printing...one in solid and the other in just block form. The first gives the sharpest printout, but the second smooths some of the blocky shape which occurs when the printout is quite large.

It is very simple to use and there are several pictures already on the disk for you to try. The only criteria is to save your instance as a full screen image even if it only takes up a small portion of that screen. You can even load two instances at a time to create interesting variations.

A chart is included which tells you how many sheets of paper a poster of each size will require and approximately how many minutes each will take to print.

I have only had the courage to print one 4X, 6X and 10X. If you use the line-up feature and position your paper as instructed, there will be a minimum of cutting and pasting to put it all together.

By printing in the block mode instead of solid, the picture could easily be colored by your children for even more fun!

This little gem costs \$15 and is distributed by Rodger Merritt of Comrodine, 1949 Evergreen Avenue, Fullerton, CA 92635. It is written by Paul Coleman in c98. Paul also wrote did Designer Labels and has done TI ARTIST and CSGD graphics.

We continue to have copies available at NorthCoast and Chips with a small commission to the clubs for each sale.

JIFFY CARD

Deanna Sheridan - NorthCoast 99ers - Cleveland, Ohio

With the arrival of Jiffy Card, you no longer need to envy your friends with Print Shop and Print Master. This program has features those other two could never perform! Jiffy Card is just about the most versatile card-making program I have come across in either the TI or IBM world.

Late last fall we reviewed JIFFY FLYER and JIFFY CARD works in the same manner. In fact, Rodger Merritt developed so many new features for JIFFY CARD, he completely rewrote JIFFY FLYER to make it more versatile also.

It loads from XB and is an XB program with assembly routines to speed up the printing process. Prompts at the bottom of the screen are used for the various functions. (Note: all the features listed here for JIFFY CARD are also available in the updated JIFFY FLYER).

It utilizes CSGD graphics and only one graphic can be used on a page. However, because you can move that graphic any place on the page and even overlay it, some interesting effects can be achieved.

Let's take a tour through the prompts:

1. DEL/INS GRAPHIC. A cursor represents diagonal corners of a box that shows the size of all CSGD graphics.

You can place this anywhere on the page, overlay, delete, etc.

2. CHANGE TITLE. You have only 10 letters for the title (which is an extra-large font), but by careful manipulation, you can create words that are much larger than 10 letters because you can copy, move and insert. For instance, "HAPPY BIRTHDAY" requires more than 10 letters. But we are in luck because several letters are repeats. If you type "BIRTHDAPPY" and copy it to a second line, then on the first line you can delete all the characters not needed for "HAPPY" and on the next line delete those not necessary for "BIRTHDAY". You will then be able to center on your card:

HAPPY

BIRTHDAY

By using this method, you can write down the side, diagonally, staggered, whatever your imagination will allow.

3. CHANGE BORDER. You page through (with the space bar) the 45 borders provided. These are the same that came with JIFFY CARD, and I think I remarked at that time, a lot of them really weren't to my liking. With this version you can create your own borders and actually save them for recall by saving them with the card you have created. I still prefer to do my artwork in SPRITEBUILDER, print out the code for the graphic and then return to JIFFY CARD. At any border, simply hit FCTM 2 and a line will appear for you to type in your hex code. The new border will immediately stay on the screen and replace the one you had previously chosen. You will, of course, lose it when you turn off your machine (or even go to another border) and the only way at this time to make it permanent is to save it with your card. I knew I would never use a lot of the borders that came with the program, so I have chosen to make my new borders permanent by typing them into the program. Simply load JIFFY CARD by calling "OLD DSK1.JIFFY CARD". Go down to about line 76 and you will see the data statements where the present borders appear. Simply type over the ones you don't want, give them new names and resave the program.

4. CHANGE CSGD GRAPHIC. Change the graphic to one of your choice.

5. ENTER TEXT. Here you have most of the editing keys of TI-Writer for delete, insert, delete line, insert, line, etc. Also, this is where you will manipulate the "BIRTHDAPPY" from the CHANGE TITLE prompt. This is also where you can type in any text you wish in one of the 7 small fonts provided.

6. CHANGE SMALL FONT. Use the space bar to choose the font you like the best.

7. SAVE, LOAD, CATALOG. Save your card to disk for later printing or editing, load a previous card, or catalog a disk.

You only see one-half of each section of your card at a time. You first create the top half of the front, then by hitting the semi-colon, it goes to the bottom half.

Graphics, TITLE and text can be used on both. To create the inside of the card, hit the '/' key and you create the top part, hit the semi-colon and create the bottom part. On the inside you do not have the option of a graphic or title, but you can use two fonts at once. The lower-case keys are one font and uppercase are another and they can be mixed. You can do some creative work here also because you have the ":()" available to draw lines and a box. You do have the option of a different border, or no border here as on the front of the card. Hit '/' again to take you back to the front of the card and you are ready to print.

Just as with JIFFY FLYER, JIFFY CARD prints out in about 2 minutes in double density format.

We have received a disk of pre-made cards by Sister Pat, and she has done a marvelous job of showing how creative one can be with this program. We have asked her permission to put it in our library so that everyone can see her work and get some ideas for cards of their own.

I feel this disk is a MUST for any Tier who likes graphics. We will continue to have this for sale at \$15 with a commission to the clubs. Rather than do a review of JIFFY FLYER, remember it prints out full-page flyers, certificates, announcements, etc. and the new version which we now have also, has exactly the same features as JIFFY CARD. JIFFY FLYER is selling for \$10.

For those out of our area, these programs can be purchased from Rodger Merritt, 1949 Evergreen Avenue, Fullerton, CA 92635 at the above prices, plus \$1.00 S&H each.

PAGE PRO 99 FROM ASGARD

Deanna Sheridan - NorthCoast 99ers - Cleveland, Ohio

If you enjoy graphics, we have a bonanza of new programs available to keep you entertained. The attendees at the Lima Conference were the first to see the newest release from Asgard, PAGE PRO 99. Evidently this started to be a form-making program similar to Rodger Merritt's FORM SHOP and ended up being "almost" a desk top publishing program.

We have struggled through various graphic printing programs never really believing that the day would come when we would have real "What You See, What You Get" (WYSIWYG) applications. Both Rodger Merritt and Asgard have changed that for us!

Page Pro 99 is written in assembly so executes rather quickly and takes some of the agony out of the frustrations we have gone through with Fontwriter, Print Wizard and others. You can save your file to disk for later printing or editing.

Page Pro has a full-page editor which toggles in the manner of TI-Writer. You are always in a graphic mode with Page Pro which means that you will be limited to 60-column width no matter what font you use. It utilizes TI-ARTIST instances and fonts which must be converted to a special format. You must be careful in converting fonts as those whose characters are not exactly the same size do not look well. In other words, stay away from those which have proportional spacing.

There is no limit to the size of the Artist instance you can use and up to 28 different pictures can be placed on

one page!. There are two fonts in residence at the same time, one large and one small. You cannot intermingle fonts. If you load in a new font, it will replace the one you have been using for the entire page. You can achieve the effect of different fonts, however, by creating some of your text in TI-Artist and loading it in as an instance. This could be done for headlines and other special effects.

There are many uses for Page Pro. It has a set of lines which lets you do invoices and forms similar to FORM SHOP. You can do two-column printing and a columizer is included. However, I had a lot of difficulty with this and ended up going another route. I prepared my text in TI-WRITER, printing it to disk in the formatter. Then I went in and divided it into two separate equal files and save them. I loaded Marty Smoley's 132-print, asked it to combine the files and print them to disk. I then had my two-columns ready to load into Page Pro. This may sound like a lot of work, but the end result looked a lot better than the columizing done with the program provided by the authors.

You can do greeting cards, again by preparing an instance in TI-Writer, turning it upside down before saving and loading into PAGE PRO. I did "THANK YOU" cards for my son to use for his graduation gifts and they turned out very nice. It would also be nice for signs, announcements, etc.

With some imagination, you can print most anything with this program, including letterheads, labels, signs, certificates, etc.

The program is very easy to use, simply sit and type as you would in TI-Writer and load your graphics at will. Keeping the graphic on the screen at all times slows you down, and you can load the graphic, then turn graphics off and a group of 'PPPPPP's' in the shape of the graphic appear so that you can still do your layout without the program continually going out to disk to pick up another section of your graphic as you work around the screen. Just turn the graphic on again when you print and it is there.

You can erase a picture or text if you are not happy with its placement. CTRL O toggles between modes. CTRL A loads another character set (large, small, or lines). CTRL L loads the pictures. CTRL F allows you to load a file or save a file or load TI-Writer text. CTRL P prints the file. Most of our first graphic printing programs only printed in single density which gives a less than desirable effect for reproduction or with colored ribbons. These later programs have rectified this deficiency and print in at least double density and sometimes more. PAGE PRO prints in both double density and quad density. Printing a page only takes 2 or 3 minutes compared to the 15 to 20 we would experience with some earlier programs.

The convert programs for instances and fonts are written in extended basic and also the columizer. Chris Bobbit stated these would be redone in assembly and provided to registered owners of PAGE PRO 99.

Incidentally, there are two versions of this program on each disk, one of the TI and the other for the GENEVE 9640. Thanks to Harry Hoffman for his graphic illustration of some of the effects he achieved with PAGE PRO 99.

If you enjoy puttering with graphics, this is another MUST program. This one is \$24.95 from Asgard Software, P.O. Box 10306, Rockville, MD 20850.






"CASLON-LG" FONT

One of the LARGE
FONTS on the
Page Pro disk.



This is "CTYPE_SM", one of the two small fonts on the program disk. There are four large fonts. You can convert any proper sized TI-ARTIST font to Page Pro format. It does take a while, (so does having a baby!), but it's well worth while to do this. I put about 64 INSTANCES on a separate disk and it really becomes easy to get used to having a lot of pics to choose from. Having more fonts will really enhance this program!

You can also use TI-ARTIST pictures (as INSTANCES), to make up headlines and special large lines.

 <p>T U L I P O 1</p>	 <p>T U L I P O 2</p>	 <p>T U L I P S O 3</p>	<p>MORE I N S T A N C E S</p>
--	---	--	---

Sometimes an experiment WILL go wrong, but can be a learning experience, and with a program like this is very easy to rectify, [at least on the screen].

The only big mistake you can make, is to SAVE to DISK using the same FILENAME as another picture you have saved! ! !

* * * * *

SPINNER

SPINNER FILENAME: DSK1.SPIN-F5018

by WESLEY R. RICHARDSON

NORTHCOAST 99ERS, CLEVELAND, OH, JULY, 1989

SPINNER is a program which gives the illusion of rotating a pattern on the screen. The options available in SPINNER are done from menu screens. Included in these options is the ability to create, modify, view, save, recall, and print SPINNER patterns. All options have an escape exit, up to the point of execution of the routine.

The pattern consists of a 32 by 32 grid of on and off positions. SPINNER will create a variety of spiral and fan patterns for you, or you can use TI-Writer to create a pattern. If you use TI-Writer, then put a space ' ' in each off position and an asterisk '*' in each on position in a 32 column by 32 row arrangement. SPINNER will create four, 90 degree rotated views of the pattern. When you SPIN the pattern, the four views are displayed sequentially. As an option, you may see all four views simultaneously.

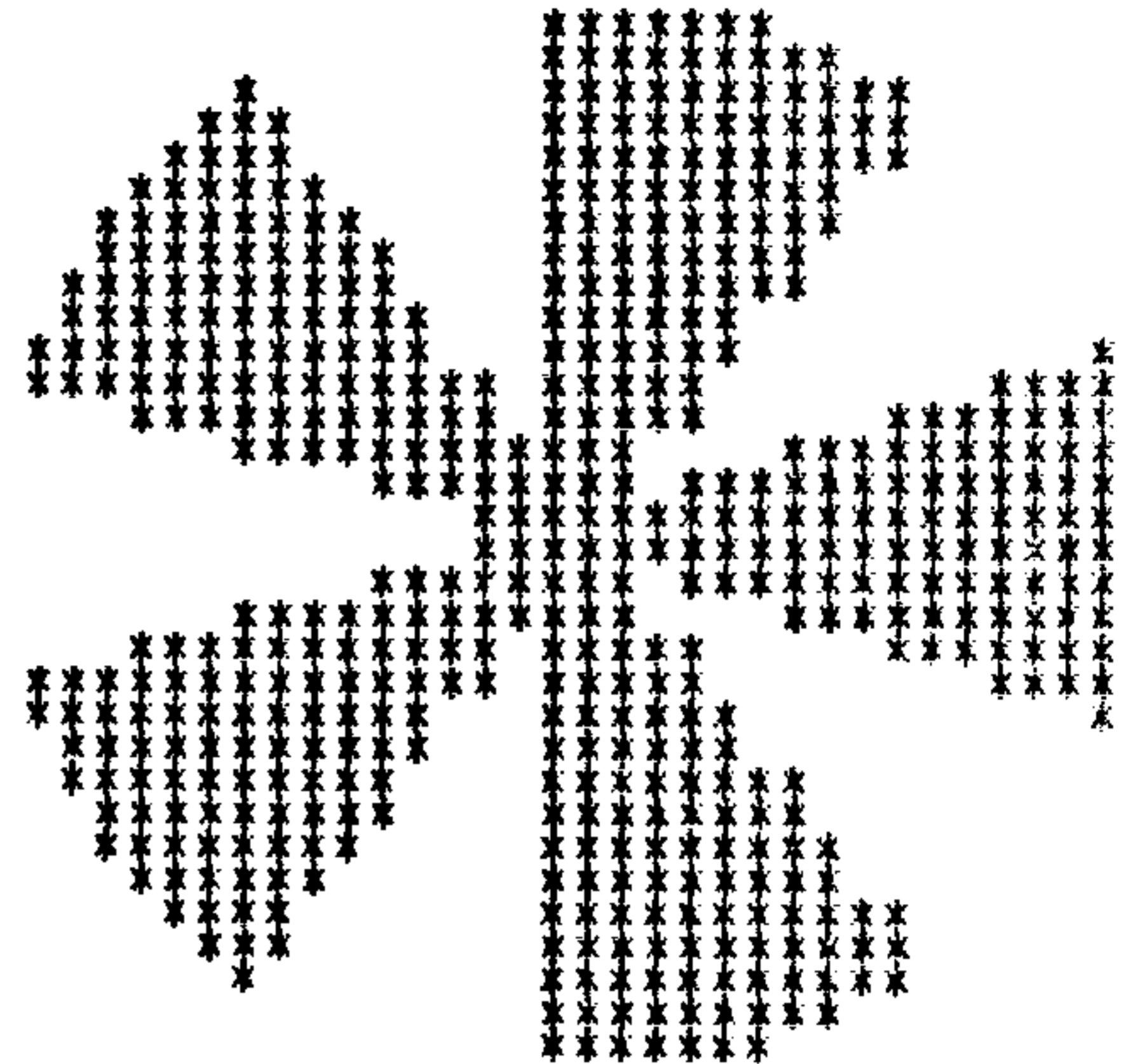
When a pattern is saved to disk, the sprite hexcodes will also be saved. If you modify the on and off positions, you do not need to change the hexcodes. The hexcodes are automatically calculated each time when the file is recalled from disk. The hexcodes correspond to the sprites in the upper left, lower left, upper right and lower right positions of each view. The hexcodes for all four views are given in order of 90 degree rotations to the right. When printing a pattern, you may print it with or without the hexcode information. The filename on the printout is the last file you used, therefore it is a good idea to save the file before printing.

The files used with this program are:

SPINNER-X Extended BASIC program
SPINNER/O Assembly object code
SPINNER/S Assembly source code
SPINNERDOC Spinner documentation

SPIN-S3018 Spinner pattern data
SPIN-F0400 Spinner pattern data
SPIN-A0001 Spinner pattern data

To enter the programs, type in the assembly source code and save the file as SPINNER/S. Then assemble the file giving SPINNER/S as the source



code and SPINNER/O as the object code. Be sure to use only the 'R' option, as Extended BASIC will not work with object files assembled with the 'C' option. Enter, save and run the Extended BASIC program SPINNER-X.

The assembly portion was written to speed up calculation of character patterns. The assembly calls include examples of passing data between Extended BASIC and assembly in the form of numbers, strings, and string arrays. To increase speed, the characters are defined in the assembly environment rather than using CALL CHAR statements from BASIC.

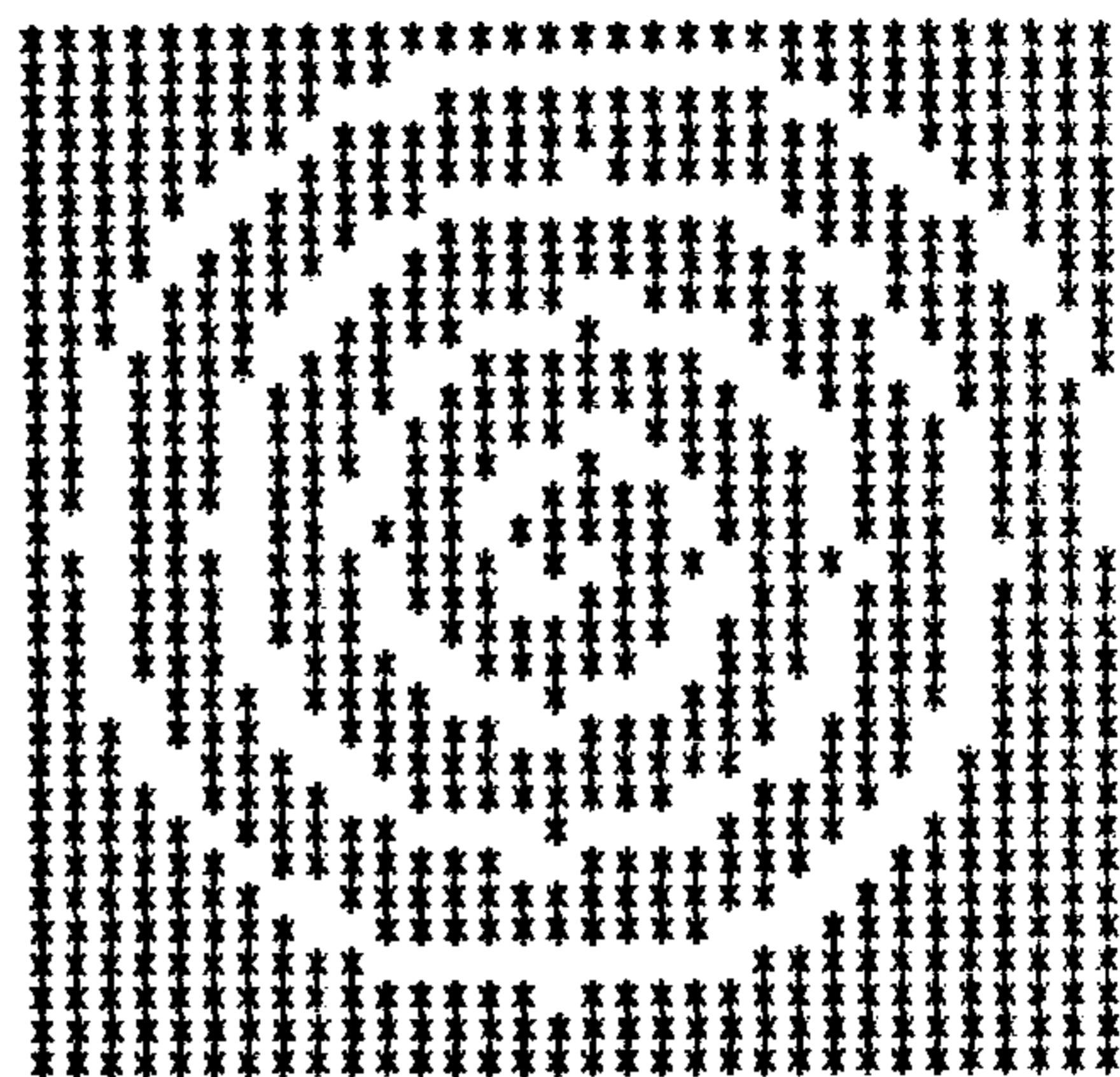
If you wish to get a copy of the files listed previously, send either a disk (SSSD or DSSD) with some of your favorite programs or copies of your club's last three newsletters to me at 27440 Pergl Road, Solon, OH 44139. I hope you enjoy SPINNER.

```
-----890614WR-----  
100 REM SPINNER-X V1.0  
110 REM WESLEY R. RICHARDSON, JUNE 198  
9  
120 REM TI-99/4A EXTENDED BASIC, CALLS  
E/A FILE DSK1.SPINNER/O CREATED F  
ROM SPINNER/S  
130 REM NORTHCOAST 99ERS, CLEVELAND, O  
H  
140 REM VARIABLES A,A1,B,BG,CA,C$(32),  
D,D$,E$,E5$,FG,G$,H$(16),I,J,K,P$,  
P1$,P2$,P3$,P4$,P5$,Q,R,S,SA,SP$,T  
,T$(5),U,V,W,W$,X,Y  
150 CALL CLEAR :: SP$='SPINNER' :: DIS
```

```

PLAY AT(6,12):SP$
160 DIM C$(32):: DIM H$(16):: DIM T$(5
)
170 GOTO 200 :: CALL CHAR :: CALL CHAR
SET :: CALL CLEAR :: CALL DELSPRIT
E :: CALL ERR :: CALL INIT :: CALL
KEY
180 CALL LINK :: CALL LOAD :: CALL MAG
NIFY :: CALL PATTERN :: CALL SCREE
N :: CALL SOUND :: CALL SPRITE
190 A,A1,B,BG,CA,C$(64),D,D$,E$,E5$,FG
,G$,H$(16),I,J,K,P$,P1$,P2$,P3$,P4
$,P5$,Q,R,S,SA,SP$,T,U,V,W,W$,X,Y
200 !@P-
210 D$="DSK1.SPIN-00001" :: P$="PIO" :
: E$=RPT$(" ",32):: E5$=" " ::
T=4
220 A1=30 :: B=PI/90 :: D=1 :: FG=16 :
: BG=5 :: G$=RPT$("F",64)
230 P1$=CHR$(27)&"E" ! TURN ON EMPHASI
ZED
240 P2$=CHR$(27)&"F" ! TURN OFF EMPHAS
IZED
250 P3$=CHR$(27)&CHR$(65)&CHR$(7)! SET
7 DOTS PER LINE
260 P4$=CHR$(27)&CHR$(65)&CHR$(12)! SE
T 12 DOTS PER LINE
270 P5$=CHR$(13)&CHR$(13)&CHR$(13)! AD
VANCE 2 LINES
280 FOR I=1 TO 32 :: C$(I)=RPT$("*",32
):: NEXT I :: FOR I=1 TO 16 :: H$(
I)=G$ :: NEXT I
290 T$(1)="DIS/FIX" :: T$(2)="DIS/VAR"
:: T$(3)="INT/FIX" :: T$(4)="INT/
VAR" :: T$(5)="PROGRAM"
300 DISPLAY AT(10,4):"REFER TO SPINNER
DOC FILE" :: DISPLAY AT(12,4):"FOR
FURTHER INSTRUCTIONS"
310 DISPLAY AT(14,4):"ON USING SPINNER
."
320 DISPLAY AT(18,4):"LOADING DSK1.SPI
NNER/0" :: CALL INIT :: CALL LOAD(
"DSK1.SPINNER/0")
330 ON ERROR 1730 :: CALL CLEAR :: DIS
PLAY AT(2,12):SP$ :: DISPLAY AT(4,
4):"BY WESLEY R. RICHARDSON"
340 DISPLAY AT(6,4):"1= CREATE NEW PAT
TERNS" :: DISPLAY AT(8,4):"2= MODI
FY IMAGE"
350 DISPLAY AT(10,4):"3= RECALL FROM D
ISK" :: DISPLAY AT(12,4):"4= SAVE
TO DISK"
360 DISPLAY AT(14,4):"5= PRINT TO PRIN
TER" :: DISPLAY AT(16,4):"6= VIEW
IMAGE"
370 DISPLAY AT(18,4):"7= CATALOG DISK"
:: DISPLAY AT(20,4):"8= END"

```



```

380 DISPLAY AT(22,4):"CHOICE? (1-8)"
390 CALL KEY(0,K,S):: IF S=0 THEN 390
:: K=K-48 :: IF (K<1)+(K>8)THEN 39
0
400 ON K GOTO 410,1030,880,960,1340,14
70,1770,1890
410 REM CREATE PATTERNS
420 CALL CLEAR :: DISPLAY AT(6,4):"CRE
ATE PATTERNS"
430 DISPLAY AT(8,4):"1= SPIRAL PATTERN
S" :: DISPLAY AT(10,4):"2= FAN PAT
TERNS"
440 DISPLAY AT(12,4):"3= OTHER" :: DIS
PLAY AT(14,4):"4= GO TO MAIN MENU"
450 DISPLAY AT(16,4):"CHOICE? (1-4)"
460 CALL KEY(0,K,S):: IF S=0 THEN 460
:: K=K-48 :: IF (K<1)+(K>4)THEN 46
0
470 ON K GOTO 480,590,790,330
480 REM CREATE SPIRAL PATTERN
490 CALL CLEAR :: DISPLAY AT(6,4):"CRE
ATE SPIRAL PATTERN"
500 DISPLAY AT(8,4):"STEP SIZE ?" :: D
ISPLAY AT(10,4):"(2.00 TO 16.00)?
"&STR$(T):: DISPLAY AT(12,4):"SIZE
=0 TO EXIT"
510 ACCEPT AT(10,21)SIZE(-5):W :: IF W
=0 THEN 330 :: IF (W<2)+(W>16)THEN
510 ELSE T=W
520 U=T/(2*PI)
530 FOR I=1 TO 32 :: C$(I)=E$ :: NEXT
I :: DISPLAY AT(12,4):2*PI
540 FOR A=PI/180 TO 2*PI STEP B :: DIS
PLAY AT(14,4):A :: CA=0.999*COS(A)
:: SA=0.999*SIN(A)

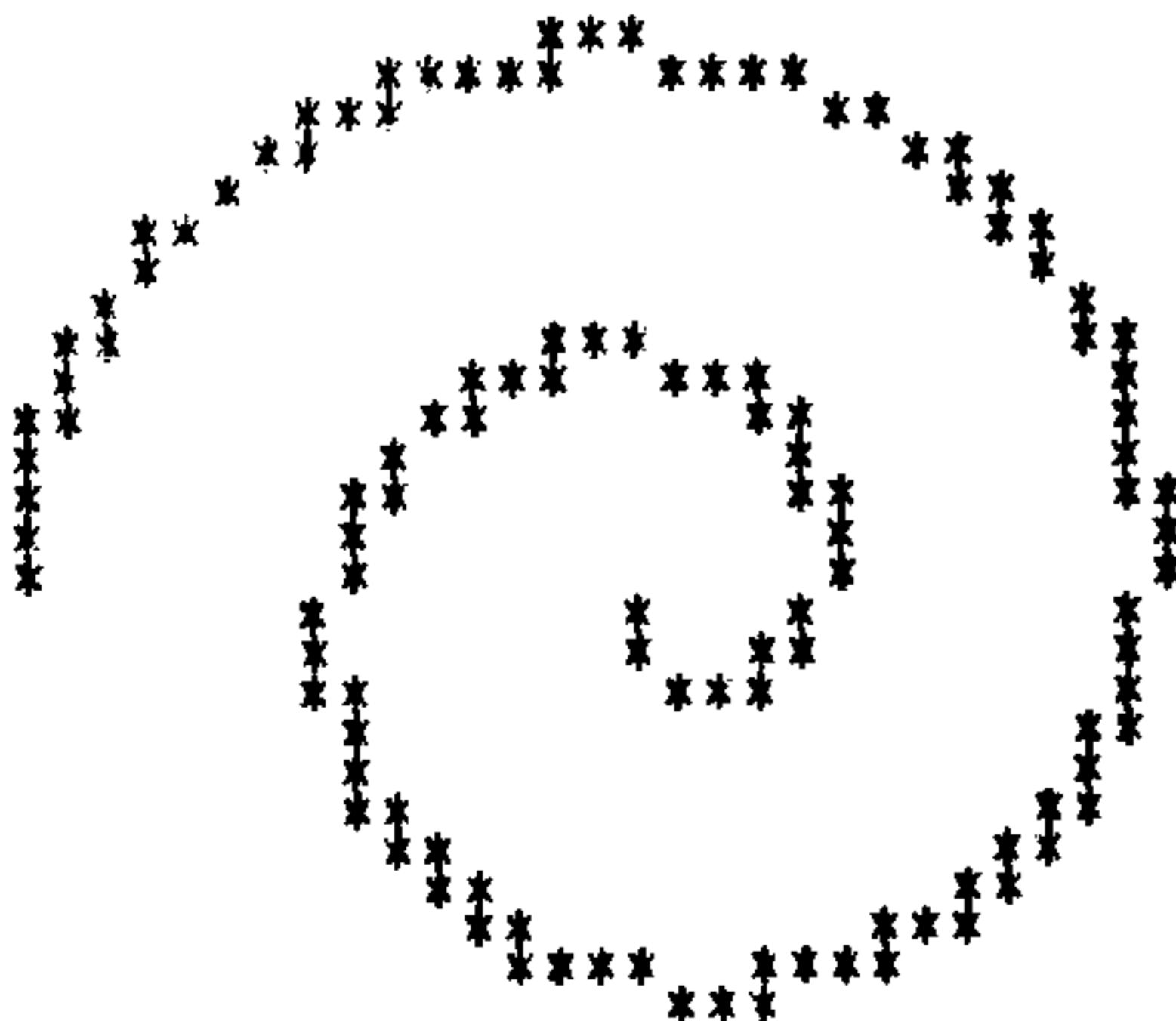
```

```

550 FOR V=U*A TO 16 STEP T
560 X=INT(V*CA+17):: Y=INT(V*SA+17)
570 CALL LINK("PUTCHR",C$( ),X,Y)
580 NEXT V :: NEXT A :: GOTO 860
590 REM CALCULATE FAN PATTERNS
600 CALL CLEAR :: DISPLAY AT(6,4):"CRE
ATE FAN PATTERN"
610 DISPLAY AT(8,4):"BLADES? (3,5,6,7)
3"
620 ACCEPT AT(8,22)SIZE(-1):K :: IF (K
=3)+(K=5)+(K=6)+(K=7)THEN 630 ELSE
620
630 DISPLAY AT(10,4):"ANGLE SIZE DEGRE
ES ?" :: DISPLAY AT(12,4):"(1 TO "
&STR$(INT(182/K))&")? "&STR$(INT(9
1/K))
640 DISPLAY AT(14,4):"ANGLE=0 TO EXIT"
650 ACCEPT AT(12,15)SIZE(-2):W :: W=IN
T(W):: IF W=0 THEN 330 :: IF (W<1)
+(W>(182/K))THEN 650 :: A1=2*INT(W
/2)+1
660 U=A1*PI/180
670 FOR I=1 TO 32 :: C$(I)=E$ :: NEXT
I :: DISPLAY AT(14,4):INT(K/2)*2*P
I/K+U+B/2
680 FOR J=0 TO K/2
690 FOR A=J*2*PI/K+U*(J<>0)-B*(J=0)/2
TO J*2*PI/K+U+B/2 STEP B
700 DISPLAY AT(16,4):A :: CA=0.999*COS
(A):: SA=0.999*SIN(A)
710 FOR V=16 TO 1 STEP -1
720 X=INT(V*CA+17):: Y=INT(V*SA+17)
730 IF SEG$(C$(Y),X,1)="*" THEN 750
740 CALL LINK("PUTCHR",C$( ),X,Y)
750 NEXT V

```

SPINNER FILENAME: DSK1.SPIN-S0B00



```

760 NEXT A :: NEXT J
770 FOR I=1 TO 16 :: C$(I)=C$(33-I)::
NEXT I
780 GOTO 860
790 REM OTHER INSTRUCTIONS
800 CALL CLEAR :: DISPLAY AT(4,6):"YOU
MAY USE TI-WRITER" :: DISPLAY AT(
6,4):"TO CREATE OTHER PATTERNS"
810 DISPLAY AT(8,4):"BY FORMING A 32 C
HARACTER" :: DISPLAY AT(10,4):"WID
E BY 32 ROWS LONG"
820 DISPLAY AT(12,4):"ARRAY OF SPACES.
" :: DISPLAY AT(14,6):"AT EACH PLA
CE WHICH"
830 DISPLAY AT(16,4):"YOU WISH TO HAVE
AN 'ON'" :: DISPLAY AT(18,4):"CHA
RACTER, PLACE AN"
840 DISPLAY AT(20,4):"ASTERISK '*.'" :
: DISPLAY AT(22,8):"PRESS ANY KEY"
850 CALL KEY(0,K,S):: IF S THEN 330 EL
SE 850
860 REM CALCULATE HEX CODES FROM EA RO
UTINES
870 CALL CLEAR :: DISPLAY AT(6,4):"GET
TING HEX CODES" :: CALL LINK("GETH
EX",C$( ),H$( )):: GOTO 330
880 REM RECALL FROM DISK
890 CALL CLEAR :: DISPLAY AT(6,4):"REC
ALL FROM DISK" :: DISPLAY AT(8,4):
D$ :: DISPLAY AT(10,4):"DSK0 TO EX
IT"
900 ACCEPT AT(8,4)SIZE(-15):W$ :: IF S
EG$(W$,1,4)="DSK0" THEN 330 :: D$=
W$ :: OPEN #1:D$,INPUT
910 FOR I=1 TO 32 :: LINPUT #1:C$(I)::
C$(I)=SEG$(C$(I),1,32):: IF EOF(1
)<>0 THEN 950
920 NEXT I :: LINPUT #1:C$(0):: IF EOF
(1)<>0 THEN 950
930 FOR I=1 TO 16 :: LINPUT #1:H$(I)::
H$(I)=SEG$(H$(I),1,64):: IF EOF(1
)<>0 THEN 950
940 NEXT I
950 CLOSE #1 :: GOTO 860
960 REM SAVE TO DISK
970 CALL CLEAR :: DISPLAY AT(6,4):"SAV
E TO DISK" :: DISPLAY AT(8,4):D$ :
: DISPLAY AT(10,4):"DSK0 TO EXIT"
980 ACCEPT AT(8,4)SIZE(-15):W$ :: IF S
EG$(W$,1,4)="DSK0" THEN 330 :: D$=
W$ :: OPEN #1:D$
990 FOR I=1 TO 32 :: PRINT #1:C$(I)&CH
R$(13):: NEXT I :: PRINT #1:CHR$(1
3)
1000 FOR I=1 TO 16 :: PRINT #1:H$(I)&CH
R$(13):: NEXT I
1010 CLOSE #1

```

...SPIN pg 4

```
1020 GOTO 330
1030 REM MODIFY IMAGE
1040 CALL CLEAR :: DISPLAY AT(6,4):"MOD
IFY IMAGE"
1050 DISPLAY AT(8,4):"1= MIRROR IMAGE"
:: DISPLAY AT(10,4):"2= INVERSE IM
AGE"
1060 DISPLAY AT(12,4):"3= TURN RIGHT 90
DEG" :: DISPLAY AT(14,4):"4= GO T
O MAIN MENU"
1070 DISPLAY AT(16,4):"CHOICE? (1-4)"
1080 CALL KEY(0,K,S):: IF S=0 THEN 1080
:: K=K-48 :: IF (K<1)+(K>4)THEN 1
080
1090 ON K GOTO 1100,1130,1160,330
1100 REM MIRROR IMAGE
1110 CALL CLEAR :: DISPLAY AT(6,4):"MIR
ROR IMAGE" :: CALL LINK("MIRROR",C
$())
1120 GOTO 860
1130 REM INVERSE IMAGE
1140 CALL CLEAR :: DISPLAY AT(6,4):"INV
ERSE IMAGE" :: CALL LINK("INVERS",
C$())
1150 GOTO 860
1160 REM TURN 90 DEG RIGHT
1170 CALL CLEAR :: DISPLAY AT(6,4):"TUR
N 90 DEG RIGHT" :: CALL LINK("RT90
",C$())
1180 GOTO 860
1190 REM FOREGROUND COLOR
1200 CALL CLEAR :: DISPLAY AT(6,4):"SET
COLORS"
1210 DISPLAY AT(8,4):"FOREGROUND (1-16)
?";FG :: DISPLAY AT(16,4):"BACKGRO
UND (1-16)?" ;BG
1220 CALL CHAR(128,G$,132,G$):: CALL MA
GNIFY(4)
1230 CALL SPRITE(#1,128,FG,72,64,#2,132
,BG,136,64)
1240 ACCEPT AT(8,23)SIZE(-2)VALIDATE(NU
MERIC," "):FG :: IF (FG<1)+(FG>16)
THEN 1240
1250 CALL SPRITE(#1,128,FG,72,64,#2,132
,BG,136,64)
1260 ACCEPT AT(16,23)SIZE(-2)VALIDATE(N
UMERIC," "):BG :: IF (BG<1)+(BG>16
)THEN 1260
1270 CALL SPRITE(#1,128,FG,72,64,#2,132
,BG,136,64)
1280 FOR I=1 TO 200 :: NEXT I :: CALL D
ELSPRITE(ALL):: GOTO 1470
1290 REM REVERSE ROTATION DIRECTION
1300 CALL CLEAR :: D=-1*D
1310 DISPLAY AT(6,4):"ROTATION DIRECTIO
N" :: DISPLAY AT(8,4):"HAS REVERSE
D TO ";D
```

```
1320 FOR I=1 TO 300 :: NEXT I
1330 GOTO 1470
1340 REM PRINT TO PRINTER
1350 CALL CLEAR :: DISPLAY AT(6,4):"PRI
NT TO PRINTER" :: DISPLAY AT(8,4):
P$ :: DISPLAY AT(10,4):"XXX TO EXI
T"
1360 ACCEPT AT(8,4)SIZE(-24):W$ :: IF S
EG$(W$,1,3)="XXX" THEN 330 :: P$=W
$
1370 DISPLAY AT(12,4):"PRINT HEX CODES
[Y/N]? N" :: ACCEPT AT(12,27)SIZE(
-1):W$
1380 OPEN #1:P$ :: PRINT #1:P1$;E5$;SP$
;" FILENAME: ";D$;P5$
1390 PRINT #1:P5$
1400 PRINT #1:P3$;
1410 FOR I=1 TO 32 :: PRINT #1:E5$&C$(I
):: NEXT I :: PRINT #1:P4$;P5$;P5$
1420 IF W$<>"Y" THEN 1450
1430 FOR I=1 TO 16 :: PRINT #1:E5$&"VIE
W"&STR$(INT((I-1)/4)+1)&" " &SH$(I)
:: NEXT I
1440 PRINT #1:CHR$(12);
1450 PRINT #1:P2$ :: CLOSE #1
1460 GOTO 330
1470 REM VIEW IMAGE
1480 CALL CLEAR :: DISPLAY AT(6,4):"VIE
W IMAGE"
1490 DISPLAY AT(8,4):"1= SPIN" :: DISPL
AY AT(10,4):"2= VIEW ALL IMAGES"
1500 DISPLAY AT(12,4):"3= REVERSE ROTAT
ION" :: DISPLAY AT(14,4):"4= CHANG
E FOREGROUND"
1510 DISPLAY AT(16,4):"5= CHANGE BACKGR
OUND" :: DISPLAY AT(18,4):"6= GO T
O MAIN MENU"
1520 DISPLAY AT(20,4):"CHOICE ? (1-6)"
1530 CALL KEY(0,K,S):: IF S=0 THEN 1530
:: K=K-48 :: IF (K<1)+(K>6)THEN 1
530
1540 ON K GOTO 1550,1650,1290,1190,1190
,330
1550 REM SPIN
1560 CALL CLEAR :: CALL MAGNIFY(4):: CA
LL SCREEN(BG)
1570 Q=104-D*24 :: R=104+D*24 :: J=16*D
1580 CALL LINK("DEFCHR",H$())
1590 CALL SPRITE(#1,80,FG,64,96,#2,84,F
G,96,96,#3,88,FG,64,128,#4,92,FG,9
6,128)
1600 FOR I=Q TO R STEP J
1610 CALL PATTERN(#1,I,#2,I+4,#3,I+8,#4
,I+12)
1620 CALL KEY(0,K,S):: IF S THEN 1640
1630 NEXT I :: GOTO 1600
1640 CALL DELSPRITE(ALL):: CALL CHARSET
```

...SPIN pg 5

```
:: CALL SCREEN(8):: GOTO 1470
1650 REM VIEW ALL IMAGES
1660 CALL CLEAR :: CALL MAGNIFY(4):: CA
LL SCREEN(BG):: CALL LINK("DEFCHR"
,H$( ))
1670 CALL SPRITE(#1,80,FG,24,56,#2,84,F
G,56,56,#3,88,FG,24,88,#4,92,FG,56
,88)
1680 CALL SPRITE(#5,96,FG,104,56,#6,100
,FG,136,56,#7,104,FG,104,88,#8,108
,FG,136,88)
1690 CALL SPRITE(#9,112,FG,24,136,#10,1
16,FG,56,136,#11,120,FG,24,168,#12
,124,FG,56,168)
1700 CALL SPRITE(#13,128,FG,104,136,#14
,132,FG,136,136,#15,136,FG,104,168
,#16,140,FG,136,168)
1710 CALL KEY(0,K,S):: IF S THEN 1720 E
LSE 1710
1720 CALL DELSPRITE(ALL):: CALL CHARSET
:: CALL SCREEN(8):: GOTO 1470
1730 REM ERROR
1740 CALL CLEAR :: CALL SCREEN(9):: DIS
PLAY AT(6,4):"ERROR ENCOUNTERED" :
: DISPLAY AT(8,4):"CODE TYPE SEVE
LINE"
1750 CALL ERR(A,I,J,K):: DISPLAY AT(10,
4):A;I;J;K :: CALL SOUND(500,110,0
)
1760 FOR I=1 TO 500 :: NEXT I :: CALL S
CREEN(8):: RETURN 330
1770 REM CATALOG DISK
1780 ON ERROR STOP :: CALL CLEAR :: DIS
PLAY AT(6,4):"CATALOG DISK (0-9) ?
" :: DISPLAY AT(8,4):"0 TO EXIT"
1790 CALL KEY(0,A,S):: IF S=0 THEN 1790
:: A=A-48 :: IF (A<0)+(A>9)THEN 1
790 :: IF A=0 THEN 330
1800 DISPLAY AT(8,17):A
1810 OPEN #1:"DSK"&STR$(A)&".",INPUT ,R
ELATIVE,INTERNAL :: INPUT #1:W$,J,
J,K
1820 PRINT "DSK"&STR$(A)&" "&W$ :: PRIN
T "AVAIL=";K;"USED=";J-K
1830 FOR A=1 TO 127 :: INPUT #1:W$,I,J,
K :: IF LEN(W$)=0 THEN 1880
1840 PRINT W$;TAB(12);J;TAB(17);T$(ABS(
I));
1850 IF ABS(I)=5 THEN 1860 ELSE C$(0)="
"&STR$(K):: PRINT SEG$(C$(0)),LEN(
C$(0))-2,3);
1860 IF I>0 THEN 1870 ELSE PRINT TAB(28
);"Y";
1870 PRINT "" :: NEXT A
1880 FOR I=1 TO 1000 :: NEXT I :: CLOSE
#1 :: GOTO 330
1890 REM END
```

```
1900 PRINT "STOP"
1910 STOP
1920 !@P+
1930 END
```

filename: SPINNER/S

```
*****
*
* SPINNER-X EXTENDED BASIC FILE *
* SPINNER/S ASSEMBLY SOURCE FILE *
* SPINNER/O ASSEMBLY OBJECT FILE *
*
* WESLEY R. RICHARDSON *
* JUNE, 1989 *
* NORTHCOAST 99ER'S - CLEVELAND, OH *
*
*****
*
DEF GETHEX,RT90,INVERS
DEF MIRROR,PUTCHR,DEFCHR
*
* EQUATES SECTION
*
NUMREF EQU >200C NUMBER GET ROUTINE
STRASG EQU >2010 STRING ASSIGN
STRREF EQU >2014 STRING GET ROUTINE
VMBW EQU >2024 VDP MULTIPLE BYTE
FAC EQU >834A FAC ADDRESS
STATUS EQU >837C BASIC STATUS REG.
GPLWS EQU >83E0 BASIC'S REGISTERS
*
* RESERVE SPACE REGISTERS AND BUFFERS
*
MYREG BSS 32 MY REGISTERS
COUT BSS 66 C$( ) CHARACTER BUF
CBUF BSS 1088 C$( ) ARRAY 32 X 34
SAVE DATA >0000 SAVE RETURN ADDR
*
*****
*
* MAIN CALL LINK SECTION *
*
*****
* GETHEX - C$( ) TO HEX CODE IN H$( )
* CALL LINK("GETHEX",C$( ),H$( ))
*
GETHEX MOV R11,@SAVE SAVE RETURN ADDR
LWPI MYREG LOAD MY REGISTERS
LI R10,1 COUNTER FOR H$( )
HLOOP BL @GETC READ C$( ) TO CBUF
BL @WRTHX WRITE HEX TO H$( )
BL @RT90C TURN C$( ) 90 DEG
CI R10,17 16 OF H$( ) DONE ?
JNE HLOOP NO, DO AGAIN
```

```

      B      @END      RETURN TO BASIC
*
* RT90 - READ C$(1) TO C$(32) INTO CBUF,
*       THEN TURN RIGHT 90 DEG.
*       CALL LINK("RT90",C$())
*
RT90  MOV   R11,@SAVE  SAVE RETURN ADDR
      LWPI  MYREG      LOAD MY REGISTERS
      BL   @GETC       READ C$() TO CBUF
      BL   @RT90C      ROTATE C$() 90 DEG
      B    @END        RETURN TO BASIC

```

```

*
* INVERS - INVERSE THE C$() PATTERN
*         CALL LINK("INVERS",C$())
*
INVERS MOV   R11,@SAVE  SAVE RETURN ADDR
      LWPI  MYREG      LOAD MY REGISTERS
      BL   @GETC       READ C$() TO CBUF
      BL   @INVERT     ON CHAR TO OFF
      B    @END        RETURN TO BASIC

```

```

*
* MIRROR - MIRROR IMAGE ABOUT VERTICAL
*         CALL LINK("MIRROR",C$())
*
MIRROR MOV   R11,@SAVE  SAVE RETURN ADDR
      LWPI  MYREG      LOAD MY REGISTERS
      BL   @GETC       READ C$() TO CBUF
      BL   @MIRROS     WRITE MIRROR C$()
      B    @END        RETURN TO BASIC

```

```

*
* PUTCHR - PUT CHARACTER C$() STRING
*         CALL LINK("PUTCHR",C$(),X,Y)
*
PUTCHR MOV   R11,@SAVE  SAVE RETURN ADDR
      LWPI  MYREG      LOAD MY REGISTERS
      BL   @PUTCHS     PUT CHAR TO STRING
      B    @END        RETURN TO BASIC

```

```

*
* DEFCHR - DEFINE CHARACTER CODES
*         CALL LINK("DEFCHR",H$())
*
DEFCHR MOV   R11,@SAVE  SAVE RETURN ADDR
      LWPI  MYREG      LOAD MY REGISTERS
      BL   @DEFCHS     DEFINE CHARACTERS
      B    @END        RETURN TO BASIC

```

```

*
* *****
* SUBROUTINE SECTION *
* *****
*
* GETC - READ C$() INTO CBUF BUFFER
*
GETC  CLR   R0          ARRAY POINTER
      LI   R1,1        VARIABLE NUMBER

```

```

LI   R2,CBUF  ADDRESS OF BUFFER
LI   R3,32    32 ELEMENTS C$()
LI   R8,>2000 C$() LENGTH = 32
LI   R9,34    ELEMENT LENGTH
GETL INC   R0    INCR ELEMENT POINT
      MOVB  R8,*R2 LENGTH OF STRING
      BLWP @STRREF READ C$() TO CBUF
      A    R9,R2  NEXT CBUF LOCATION
      DEC  R3     DONE 32 ELEMENTS?
      JNE  GETL   NO, THEN LOOP
      B    *R11  RETURN

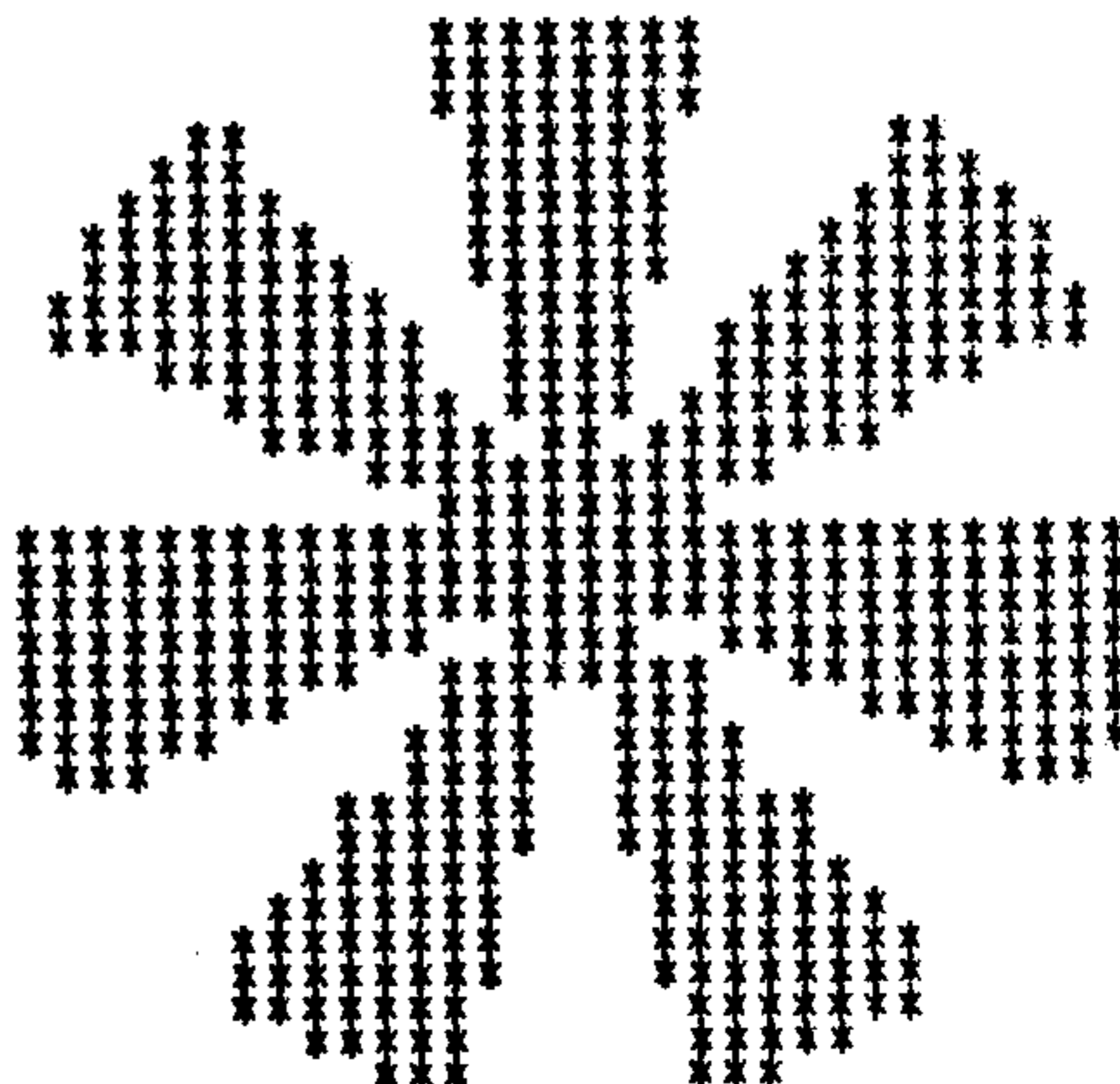
```

```

*
* WRTHEX - WRITE HEX PATTERNS TO H$()
*
WRTHEX LI   R7,>2A00  ON CHARACTER
      CLR   R8        R8=COL 0-32 INC 16
HEX8   CLR   R9        R9=ROW 0-1088 +544
HEX9   CLR   R13       R13=COL 0-16 INC 8
      LI   R5,COUT     COUT POINTER
      LI   R0,>4000    LENGTH H$() =64
      MOVB R0,*R5     STO LENGTH IN COUT
      INC  R5        POINT FIRST BYTE
HEX13  CLR   R14       R14=ROW 0-544 INC 34
HEX14  CLR   R15       R15=CHAR L-R 0-8 + 4
HEX15  CLR   R1        R1 =CHAR BYTE BUFF
      CLR   R2        R2 =BUILD BYTE
      LI   R4,>0800    BIT MASK
      CLR   R6        R6=CHAR 0-3 INCR 1
HEX6   LI   R0,CBUF    R0=CBUF POINTER
      INC  R0         R0=R0 + 1
      A    R8,R0      R0=R0 + R8
      A    R9,R0      R0=R0 + R9
      A    R13,R0     R0=R0 + R13
      A    R14,R0     R0=R0 + R14
      A    R15,R0     R0=R0 + R15

```

SPINNER FILENAME: DSK1.SPIN-F7013



	A	R6,R0	R0=R0 + R6	A	R0,R7	ELEMENT + POSITION	
	MOVB	*R0,R1	GET CBUF BYTE	MOVB	R8,*R2	STO LENGTH STRING	
	CB	R1,R7	IS CHAR ON?	LI	R4,32	CHARACTERS TO GET	
	JNE	HEXOFF	JUMP IF CHAR OFF	MOV	R2,R5	COUT POINTER	
	SOCB	R4,R2	TURN BIT ON	RT90L2	INC	R5	INC COUT BUFFER
HEXOFF	SRL	R4,1	SHIFT MASK 1 PLACE	MOVB	*R7,*R5	MOVE CHAR TO CBUF	
	INC	R6	INC BIT POINTER	S	R9,R7	DEC POINTER 1 ROW	
	CI	R6,4	DONE WITH 4 BITS?	DEC	R4	MOVED 32 CHARS?	
	JNE	HEX6	JUMP IF NOT DONE	JNE	RT90L2	NO, THEN DO AGAIN	
*				BLWP	@STRASG	WRITE C\$() ARRAY	
	CI	R2,>0A00	CHAR >= 10 ?	DEC	R3	WRITE 32 ELEMENTS?	
	JL	HEXLOW	JUMP IF <10	JNE	RT90L1	NO, THEN DO AGAIN	
	AI	R2,>0700	ADJUST FOR ABCDEF	B	*R11	RETURN	
HEXLOW	AI	R2,>3000	CONV TO STRING VAL	*			
	MOVB	R2,*R5+	STORE COUT INCR R5	* INVERT - ON CHAR TO OFF, OFF TO ON			
	AI	R15,4	INCR COUNTER	*			
	CI	R15,8	IS LOOP DONE?	INVERT	CLR	R0	ARRAY ELEMENT
	JNE	HEX15	JUMP IF NOT DONE		LI	R1,1	VARIABLE FOR C\$()
*					LI	R2,COUT	C\$() BUILD BUFFER
	AI	R14,34	INCR COUNTER		CLR	R3	TEMP STORAGE CHAR
	CI	R14,544	IS LOOP DONE?		CLR	R6	ROW IN CBUF
	JNE	HEX14	JUMP IF NOT DONE		LI	R8,>2000	CHAR OFF " "
*					LI	R9,>2A00	CHAR ON "*"
	AI	R13,8	INCR COUNTER	INVLP1	INC	R0	POINT TO C\$()
	CI	R13,16	IS LOOP DONE?		MOVB	R8,*R2	LENGTH STRING=32
	JNE	HEX13	JUMP IF NOT DONE		LI	R4,32	32 CHAR TO GET
*					LI	R5,COUT	POINTER FOR COUT
	MOV	R10,R0	H\$() ELEMENT POINT		LI	R7,CBUF	POINTER FOR CBUF
	LI	R1,2	VARIABLE NUMBER X8		A	R6,R7	ADD ROW TO POINTER
	LI	R2,COUT	H\$() BUILD BUFFER	INVLP2	INC	R5	ADD COL TO POINTER
	BLWP	@STRASG	WRITE H\$() ARRAY		INC	R7	ADD COL TO POINTER
	INC	R10	NEXT H\$() ELEMENT		MOVB	R9,*R5	STORE "ON" CHAR
	AI	R9,544	INC COUNT 16 X 34		MOVB	*R7,R3	GET CHAR FROM CBUF
	CI	R9,1088	IS LOOP DONE?		CB	R3,R9	IS CHAR ON?
	JNE	HEX9	JUMP IF NOT DONE		JNE	INVOFF	JUMP IF CHAR OFF
*					MOVB	R8,*R5	STORE "OFF" CHAR
	AI	R8,16	INCR COUNTER	INVOFF	DEC	R4	DONE WITH 32 CHAR?
	CI	R8,32	IS LOOP DONE?		JNE	INVLP2	NO, DO AGAIN
	JNE	HEX8	JUMP IF NOT DONE		BLWP	@STRASG	YES, WRITE TO C\$()
*					AI	R6,34	GO TO NEXT ROW
	B	*R11	RETURN		CI	R0,32	32 ELEMENTS OUT?
*					JNE	INVLP1	NO, DO AGAIN
					B	*R11	YES, RETURN
*				*			
				* MIRROS - MIRROR IMAGE ABOUT VERTICAL			
				*			
* RT90C - TURN C\$() 90 DEG RIGHT,							
* AND WRITE TO ARRAY.							
RT90C	CLR	R0	ARRAY ELEMENT	MIRROS	CLR	R0	ARRAY POINTER
	LI	R1,1	VARIABLE NUMBER		LI	R1,1	VARIABLE NUMB C\$()
	LI	R2,COUT	C\$() BUILD BUFFER		LI	R2,COUT	C\$() BUILD BUFFER
	LI	R3,32	NUMB ELEMENTS C\$()		CLR	R6	ROW IN CBUF
	LI	R6,1054	POINT C\$(32) 31X34		LI	R8,>2000	LENGTH STRING=32
	LI	R8,>2000	LENGTH IN C\$()=32	MIRLP1	INC	R0	POINT TO C\$()
	LI	R9,34	BUF ELEMENT LENGTH		MOVB	R8,*R2	STO LENGTH STR =32
RT90L1	INC	R0	INC ELEMENT POINT		LI	R4,32	32 CHAR TO GET
	LI	R7,CBUF	ADDR OF C\$() BUFF		LI	R5,COUT	POINTER FOR COUT
	A	R6,R7	SET R7 TO C\$(32)		LI	R7,CBUF	POINTER FOR CBUF

A FIX FOR STUTTERING KEYS ON THE TI-99/4A
By Glenn Bernasek
TI-Chips Cleveland

Have you ever been typing along on your 99/4A, without a care in the world, and looked back on the screen to find words spelled with too many E's, I's, A's or what ever? It's a real pain in the you-know-what isn't it? Now you have to go back and edit everything that you just typed. A frustrating time waster!

Here's a simple procedure that might prevent or at least alleviate the problem of the "stuttering" keys.

The key pad caps, on MOST of the ORIGINAL TI-99/4A and Radio Shack replacement keyboards, pop off. Inside the SQUARE key stems are, what I call, "clam-shell" contacts. These contacts are made of a very thin, gold plated, spring metal, and are designed to be "bounce" resistant. (It's the bounce that causes most of the key "stuttering".)

If these contacts are very dirty, they will make an intermittent contact when closed, therefore resulting in un-wanted additional characters due to "bounce". (No matter how you press a "stuttering" key, it will eventually produce repetitive characters.)

What I did to make my TI keyboard as solid as the "Rock of Gibraltar" again, was to pop off all of the key pad caps, and clean the "clam-shell" contacts with a 1/4 inch wide

strip of VERY FINE grade fingernail emeryboard wetted with some rubbing alcohol. I just lightly push on the center dividing bar, between the contacts, two or three times with the alcohol damp emeryboard. This washed and polished the contact surfaces, and no additional rinsing or drying was necessary.

The alcohol was allowed to evaporate before I pressed the PROPER key pad caps back into their respective positions. (BY THE WAY, BE VERY CAREFUL WHEN INSERTING AND MOVING THE EMERYBOARD ... THE CONTACTS ARE MADE OF VERY THIN MATERIAL.)

To keep your keyboard in good working condition and minimizing corrective maintenance, just follow these three simple precautions:

1. Do not eat, drink or allow your family pets near the keyboard.
2. Ashes and tars from tobacco smoke will coat electrical contacts just as easily as they coat the lungs. Don't smoke near the computer at any time.
3. An unprotected keyboard is open for anything. Keep it covered when not in use.

CLEVELAND AREA 99/4A USERS GROUPS
C/O DEANNA SHERIDAN
20311 LAKE ROAD
ROCKY RIVER, OH 44116

CHECK YOUR EXPIRATION DATE.
THIS MAY BE YOUR LAST ISSUE!

Exp Date: 89/07

!! TIME DATED MATERIAL !!

