THE GUILFORD 99'ER NEWSLETTER

VOL.6 NO.7 JULY 1989

Scott Hughes, Pres.(227-2688) Emmett Hughes, Vice Pres.(584-5108)
Mack Jones, Secy/Treas(288-4280) Herman Geschwind,Pgm/Library(288-0015)
BBS: (919)274-7000(OPUS) ROS: (919)621-2623

The Guilford 99'er Users' Group Newsletter is free to dues paying members (One copy per family, please). Dues are $12.00 per family, per year. Send check to 3202 Canterbury Dr., Greensboro, NC 27408. The Software Library is for dues paying members only. (George von Seth, Ed.:292-2035)

OUR NEXT MEETING

DATE: July 5, 1989 Time: 7:30 PM. Place: Glenwood Recreation Center, 2010 S. Chapman Street.

Program for this meeting will be a demonstration of the ADVENTURE EDITOR from TEXCOMP. Mack Jones will show you how to write your own adventure using this software package. Stop by for a look at his GRAVEYARD ADVENTURE!!

MINUTES

The June 6th meeting of the Guilford 99er Users' Group was held at the Glenwood Recreation Center in Greensboro, N.C. There were 11 members present.

The meeting was called to order by V.P. Emmet Hughes at 7:38 P.M. but was turned over to Pres. Scott Hughes as soon as he arrived a few minutes later.

Old Business:

a. Scott asked if the McInker would handle reel to reel type ribbons and the answer was no. At the present time, all we have are provisions for inking the cartridge type ribbons. It was also made note that we need a shaft with a left hand screw to handle clockwise turning ribbons.

New Business:

a. Tony Kleen announced that a friend was willing to sell a complete TI outfit for $200. Tony was asked to see if his friend would be willing to sell components seperately.

b. Scott brought up the fact that the TI Echo was no longer being carried by the "Backbone" and asked if anyone knew the reason. He also mentioned that it would be carried on a Raleigh BBS.

c. Emmet brought a console and several items that a friend wished to sell, but since the console was defective and there were no prices put on the other items, there were no takers.

d. Members were asked to demo something for meetings and yours truly volunteered to do the July program. It will be a demo on the Adventure Editor and Adventure Programming Language (APL).

e. Since the July meeting would normally fall on the 4th, members elected to meet Wednesday the 5th instead. This will be for the July meeting only.

Bob Carmany gave a demo on using Disk Utilities to repair "blown" disks. It was a very good demo. Bob deliberately destroyed a disk starting with sector 0 and then showed how to repair it. He then did the same with sector 1 and repaired it which is quite a bit different chore than sector 0. Bob received a hand from the group at the conclusion. Thanks Bob.

The meeting was adjourned at 9:15 P.M.

Respectfully Submitted,
L.F. "Mac" Jones,Sec/Treas.
Guilford 99er Users' Group

RAMBLING BYTES

by "Mac"

Well, if I can get something written between thunder storms, I will finish this so Bob and George can piece the newsletter together! It seems just as I get in the notion to do a little typing, another burst of thunder announces the arrival of another storm. I heard something laughing out in the yard and looked out and my grass was really laughing! Two mowings a week is getting to be a little much, Grin.

If I may, I would like to mention something that I know is in the back of most older folks' mind, but we just don't like to think about it. Yep, it's the shirking off this mortal coil, as Shakespeare so amply put it, and look to the big TI in the sky. My reason for mentioning this is simply this....a few years ago, I had a good friend who probably thought, like me until then, that he would be around for a long time. So, he never got around to making out a will. Upon his untimely demise, his two boys of course thought that everything their pop had was automatically theirs. Not so! They had to buy certain things that they wanted from the State, and relatives they had never heard of got shares of the estate that they felt, and rightly so, were theirs. You had better believe that it put the fear into me. I had recently installed a large key-telephone system for a pair of lawyers new to Greensboro and as the system took several days to finish I had become pretty friendly with them and they with me. I called them and made an appointment for my wife and me for the next week.

I really don't know the cost now, but in the early years of 1980 it was not very expensive for us. Now I can rest assured that my wife and three girls will get what I have instead of the State of N.C.!

Just get you a piece of paper and a pen/pencil and crawl in that easy chair and list all the TI equipment you own piece by piece. Put in the fair amount that you could expect to get for each piece and also what you think the fair price of the whole kit and kiboodle could bring and put all of this into an envelope with the contents written on the envelope. You can't imagine what a lot of trouble this would save your adminstrator if some idiot should cross the median and swat you head-on. And it doesn't always happen to someone else as we like to believe! I hope this hasn't been too morbid for you but if I can save just one of you the heartache that I witnessed that day, then I feel it has been worth every word.

To get on a brighter note (?), we have been missing a few members and we sure would like to see them again soon. Hopefully, for the July meet, everyone can turn out. I realize that July is a lot of folks' vacation time but just one evening wouldn't dent it that bad if they were home anyhow and not at the beach or the mountains.

While reading the Houston newsletter I notice that they are having a bunch of trouble out there with Ma Bell. Seems as how Ma is wanting to provide the BBS for the users and cutting out the SySops' or making it so expensive for them that they will want to throw in the towel. They warn the Eastern users to be on the lookout for it to hit us also. Seems Ma is calling it "Information System". I hope it doesn't end our ROS Groundstar BBS, as that is the only TI board that we have in Greensboro.

It's growing darker by the minute and I can see the flashing getting closer so I had better knock it off. Even tho I have spike-arrestors on every one of my components, I still don't like to take the chance of a blown computer so hang in there and see ya at the July meet. Until then enjoy the good TImes.

EXTENDING XB

One of the better things that TI did when they designed the Extended BASIC cartridge was to add a "hook" to load and access Assembly Language programs from the XB environment. This was accomplished with CALL LINK. Many years ago, Craig Miller started everything with a few LINKed routines like VPEEK and VPOKE that allowed one to view and load values directly into VDP memory. From these meager beginnings, the XB environment was expanded and stretched.

One of the first commercial entires to take advantage of LINKed routines was the AMERISOFT graphics package. There are still copies of this now discontinued package floating around. It created 40+ new routines that could be accessed from within an XB program. There were windows, graphics drawing routines, screen dumps, and other easily-used A/L routines. Most were based on the European APESOFT graphics. This was a significant step forward.

Over the years, several other extensions of the XB environment have appeared in the "fairware" market. XXB by Barry Traver et. al. has been revised a couple of times and now provides a number of utilities in both 40-column and 32-column mode. EDP (Enhancement Display Package) is another utility that does much the same thing. There are improved versions of the XB DISPLAY AT and ACCEPT AT commands as well as other welcome additions to the XB environment. XDP (eXtended Display Package) is another one that stretches the XB environment to limits previously not possible.

Basically, with the exception of AMERISOFT, the XB utilities are very similar. One will find a good deal of overlap between all of them. There are just enough differences to pique a person's curiosity, though. You will invariable find a couple of interesting routines in one of the packages that are missing from the others. XXB, for example, has a MERGE file that eliminates the necessity for CALL LINK entirely. For a slight sacrifice in speed, you can type the commands in as if they were XB reserved words. Thus, CALL LINK("BYE") simply becomes CALL BYE.

The best thing to do is get a copy of these software packages and try them out for yourself. Most are up on ROS and can be easily downloaded and used. You will be amazed at what you can do with them!!

NEW STUFF

As summer starts, programmers tend to do things other than program. Of course, that leads to a shortage of new software packages for our beloved TI. There are a few that continue to trickle in. Some of them are really new and others are new discoveries.

Charles Earl has introduced HOTBUG -- a "fairware" de-bugger that operates in two modes. It is a single step debugger as well as a run-time debugger. One of the more interesting features in the REMOTE option that allows the disassembly and debugging of an A/L program by modem. Very interesting, indeed!!

One of the new discoveries is 1000 WORDS by Norman Rokke. This program is a conversion for TI-ARTIST pics that changes them to a format readable by the TI-WRITER (or FUNNELWEB) Formatter. That makes it easier to incorporate graphics into your text presentations. The program comes with instructions and is relatively easy to use. The previously mentioned XXB (Vn 1.5) and XDP packages are also recent releases as well.

If you missed the last issue of MICROpendium, you missed the first of the XB utilities for Vn 2.x of TI-BASE. There is a short program to read your disk directories and convert them into a format that can be used by TI-BASE. It is really quite crude and limited, but it is a start nevertheless. Let's hope that some more programs of that kind make an appearance in the near future.

ASSEMBLY LANGUAGE SQUEEZING PART 1

By Tony McGovern
Funnelweb Farm

When you first start writing assembly code you find that initially there is a vast arena to work in. As projects get more ambitious the crunch eventually comes, or you may have to squeeze some code into a gap that just seems too small at first. Then it is time to scrunch down your code to fit. In this series of short articles we will look at various tricks that can be used to save bytes or words here or there. There are larger issues to be consedered, of program organization and identification of repetitive code segments, but we'll just play around on the microscopic scale.

For our first topic let's look at the problems of advancing register pointers. This is often handled directly by auto-increment addressing.

```
MOV *R0+,*R1+
MOVB R0+,*R1
```

Now suppose you want to advance the pointer by 4. The obvious solution is to use

```
AI R0,4
```

This takes two words or 4 bytes. A shorter solution is to use

```
C *R0+,*R0+
```

This gets the job done in only 1 word, saving 2 bytes on each use. The comparison C only reads the data pointed to, and you just ignore any setting of comparison flags. Like most things that shorten code, it takes more machine cycles to execute, but this would hardly matter except perhaps in the innermost heart of a frequently executed loop. One little trap in the TI-99/4A is that some memory mapped devices respond even to a read from their address, so you should be sure that R0 does not point to such an address. This is rarely a problem though.

It is possible to come up with some more littl byte-savers along these lines. Sometimes you have to keep up with incrementing two register pointers simultaneously, and the code is such that auto-incrementing is not done. This often occurs when maintaining pointers to addresses in GROM or VDP RAM. Suppose their pointers are in registers R8 and R9 and both must be incremented. The obvious coding is

```
INC R8
INC R9
```

which takes 2 words. Alert readers will already have guessed a shorter method

```
CB *R8+,*R9+
```

to increment both in one word of code. So if you come across strange comparisons being made, remember that comparison may not have been the intention.

THE WIZARD(S) OF OZ

It is really amazing how some places seem to be "hot spots" for programmers. One of the most consistantly innovative areas in the entire TI world has been OZ ---Australia. The relative isolation of the island continent has sparked programming by necessity. Consider, if you will, the Hunter Valley UG.

Years ago, a father and son team came up with a hybrid XB and A/L program that very neatly prints out a 4-column listing of either A/L source code or an XB program. The only preparation is that the program be LISTed to disk before the program is run. COLIST was an immediate success and, to this day, is in a class by itself. Tony and Will McGovern were soon to be introduced to the TI community in a VERY big way.

After COLIST, Tony and Will came up with perhaps the most significant program ever written for the TI ---- FUNLWRITER. When the sole supplier of the TI-Writer and E/A cartridges raised prices to an astronomical level, the McGoverns responded with a TI-Writer loader ---sans cartridge. FUNLWRITER grew and evolved into FUNNELWEB which is the most versatile programming shell ever written for the TI. It is doubtful that many of the programs written in the past couple of years would have been attempted at all if not for the inspiration of the McGoverns with FUNLWRITER. They have also produced and released many of the developmental programs that they used throughout the years as "fairware" and each and every one of their releases is truly excellent.

One such duo of programmers would be enough to make any UG famous but the sheer raw talent doesn't stop there. One of the latest entries from "down under" is the QED 4.4 loader. This little gem from Ron Kleinschafer was originally designed to load FUNNELWEB and other programs into the QED 32K cartridge designed by Neil Quigg. One thing led to another and the latest version will load these same programs into a SuperCart as well. Even loading a customized version of FUNNELWEB has been made as painless as possible. Ron's programming is just as professional as Will and Tony's. The amazing part of the whole situation is that Ron is a completely self-taught A/L programmer. An Opal miner by trade, he lives so far out in the outback that the kangaroos get lost there. His philosophy ---- all you can do is wreck it!!

The number of other projects produced by the Hunter Valley UG are simply staggering. Looking through back issues of the newsletter is like a trip through a TI "wish list". There are hardware projects to build a 32K for the console (long before the idea came to the U.S.), a 3-slot PEB, and sundry and assorted console modifications. Neil Quigg, who is closely associated with the HV 99'ers has come up with the 32K QED cartridge that combines all of the best attributes of the SuperCart, and the QUEST 200 RAMdisk --- a full 512K of usable superfast RAMdisk. The utilities, of course are being written by Tony McGovern.

It boggles the mind to think that this UG with 100+ members has produced so much for the TI community. What else is there? How about an 80-column TI-Writer editor for the AVPC card (Tony McGovern). Should you want to see just what they are up to, ask Mac to look at the newsletters at the next meeting ---they are "The Wizards of Oz".

ADVENTURING

Since Mac was kind enough to volunteer to do a demo of the ADVENTURE EDITOR at this meeting, maybe a few general tips are in order about playing adventure games. First of all, there are two major groups of adventure games that have been written for the TI. There are the Scott Adams games that require the TI ADVENTURE cartridge and the INFOCOM games which do not and are the most difficult of the two groups.

The first things that you will need when you play an adventure game are a piece of paper (large) and a pencil. These are needed for drawing your adventure map. Since these games can get quite complicated, you will have to write down where you have been and what is there. The best way to do this is to draw a map showing each of the rooms, what objects are in each, and how they connect to other rooms. "Rooms" are either physical rooms or are locations that you can go to during the course of the adventure. Why is this important? Because you will probably get killed the first few times that you play even the simplest of these games. If you keep track of what you have done and where you have been, you will be able to get progressively further into the game until you are eventually able to solve it.

Certainly, some of the simpler Scott Adams games can been solved in the course of a rainy Sunday afternoon but most are several-day affairs. When you start dealing with the INFOCOM games like HITCHHIKER'S GUIDE TO THE GALAXY, you can spend weeks or even months seeking a resolution to the game.

To play an adventure game, you must be curious. Look at, prod, pick up, and carry everything that you can the first time through. You never know what you might run across that might be of value later on in the game and often times you can't go back and pick it up when you need it.

Finally, if you get into trouble, there are hint books and "cheat files" for all of the Scott Adams adventures and most of the INFOCOM games. So one rainy afternoon test your wits in one of these adventure games and when you are through with them all ---- write your own like Mac is doing.

PROGRAMS! PROGRAMS!

Here are a couple of programs that might be of interest to you. The first one is a game called METEOR RESCUE that has been passed on from UG to UG. Unfortunately, the author's attribution has been lost in the process. The game is a very simple one and the instructions are built-in. Type it in and you will find that it is simple and at the same time just a bit frustrating to play. If anyone knows who authored this program, please contact Bob Carmany so that the attribution can be restored and proper credit given.

The second program is a utility to be used with your Speech Synthesizer. The program is explained fully in the remarks at the beginning. It can be SAVEd as a MERGE program and then MERGEd into the appropriate program whenever you need it. Enjoy!!!

```
10 !//////////////////
20 !/ METEOR RESCUE /
30 !//////////////////
40 !
50 !///INITIALIZING///
60 CALL MAGNIFY(3):: CALL CL
EAR :: DIM DI(6)
70 CALL CHARPAT(96,A$):: IF
POS(A$,"02B0D",1)THEN 220

80 !/LANDING MODULE/
90 DATA 00002B0D1D1F1F053F23
204040E000002810D0F0F8F8F8A0
FCC404020207

100 !/MAN/
110 DATA 101438501828282,105
0381430282808

120 !/ASTEROIDS/
130 DATA 00000000181F0F1B1B3
F1F1F04000000000000C7CD8F8F
0F874F8F830,00000000090F070D
05050707010000000000000080C0
F8E4F8B8FC3EFE3E02
140 DATA 000001020307070707B0
B1E0E0D0E0E0440C0C0C0E0E0E0B
8F8E0C0808
150 DATA 0018270F1B0A17694F2
070F66FE77FFE50404082060A48C
8DFEDCF766DC774
160 DATA 0000000000000000000
00103020100010000000000000000
00000B040C0008
170 DATA 0000C995DD95D500000
0000000000000000D312F352DB0
0
180 DATA FEFFFFFFFFFFFFFF010
91D3F37230102000080C0E0F0F8F
C80C89CFEF6E2C0A0
190 DATA FF000000000000000000
000000000000FF
200 FOR I=96 TO 136 STEP 4 :
: READ A$ :: CALL CHAR(I,A$&
RPT$("0",64-LEN(A$))):: NEXT
 I

210 !///SCREEN SET UP///
220 GOSUB 690 :: CALL SCREEN
(2)
230 CALL CLEAR :: FOR I=1 TO
 30 :: CALL VCHAR(RND*19+2,R
ND*29+2,43):: NEXT I
```

```
240 CALL CHAR(47,"48386C6C38
4482",41,"FEFFF1E0E0C0C080"&
RPT$("F",16)&"1",140,RPT$("F
",64))

250 !//SET UP METEOR//
260 H=-1 :: CALL INIT :: CAL
L LOAD(-31878,16)
270 FOR I=56 TO 152 STEP 16
:: RANDOMIZE
280 F=INT(RND*3-1):: IF F TH
EN DI(INT((H+2)/2))=F :: F=F
*INT(RND*SP+SP-3)ELSE 280
290 RANDOMIZE :: H=H+2 :: CA
LL SPRITE(#H,INT(RND*3+1)*4+
104,RND*5+10,I-14,252,#H+1,I
NT(RND*3+1)*4+104,RND*5+10,I
-14,RND*200+25):: NEXT I
300 GOSUB 670

310 !//GROUND//
320 FOR J=1 TO 32 :: CALL VC
HAR(24,J,INT(RND*12+108))::
NEXT J
330 CALL VCHAR(22,16,119,2):
: CALL VCHAR(22,15,114):: CA
LL VCHAR(22,17,116)
340 CALL HCHAR(4,1,42,5):: C
ALL VCHAR(4,6,41):: CALL VCH
AR(1,1,143,2):: CALL VCHAR(1
,2,143,3)
350 CALL HCHAR(1,22,47,2-SH)
360 MS=184 :: HI=12 :: FOR H
=28 TO 19 STEP -2 :: MS=MS+1
 :: FOR F=0 TO 1 :: CALL SPR
ITE(#H-F,136,1,MS,HI):: HI=H
I+26 :: NEXT F :: NEXT H
370 CALL COLOR(1,16,1,2,16,1
,10,12,1,11,1,12,12,15,1,14,
7,1,9,7,1):: FOR I=3 TO 8 ::
 CALL COLOR(I,16,1):: NEXT I
380 CALL SPRITE(#15,96,15,11
,30,#17,128,3,24,26,#18,132,
7,17,10,#24,140,1,171,116)
390 HI,MS,F,H=0

400 !///GAME LOOP///
410 CALL KEY(1,X,Y):: IF Y T
HEN 470 ELSE CALL JOYST(1,Y,
X):: IF X=0 AND Y=0 THEN 430
 ELSE XVEL=-X :: YVEL=Y :: C
ALL SOUND(300,110,18,-4,20)
420 CALL MOTION(#15,XVEL,YVE
L)
```

```
430 CALL COINC(ALL,HIT):: IF
 HIT THEN 440 ELSE 410
440 CALL SOUND(200,-5,0):: H
I=HI+1 :: IF HI>2 THEN CALL
BLOW(SH,MS):: CALL HCHAR(1,2
2,32,9):: IF SH>2 THEN 540 E
LSE HI,MS=0 :: CALL HCHAR(1,
22,47,2-SH)
450 GOTO 410

460 !//LAND?//
470 CALL MOTION(#15,0,0):: C
ALL SOUND(500,-5,20):: CALL
COINC(#15,155,117,1,HIT):: C
ALL POSITION(#15,X,Y):: CALL
 SPRITE(#16,124,10,X,Y)
480 CALL COINC(ALL,HIT2):: I
F HITZ THEN CALL DELSPRITE(#
16):: GOTO 440 ELSE CALL COI
NC(#15,11,31,1,HIT2):: IF HI
T2 AND MS THEN CALL LS :: F=
F+1 :: HI,MS=0 ELSE 510
490 IF F=10 THEN 550 ELSE IF
 F/2=INT(F/2)THEN GOSUB 670
:: SH=SH-1 :: CALL HCHAR(1,2
2,47,2-SH)
500 GOTO 520
510 IF HIT=0 OR MS=1 THEN CA
LL DELSPRITE(#16):: GOTO 410
 ELSE CALL LAND :: MS=1
520 XVEL,YVEL=0 :: CALL HCHA
R(1,9,100,F):: GOTO 410

530 !//END MESSAGES//
540 Y$="YOU HAVE FAILED. YOU
'LL HAVE TO LEAVE THE REST H
ERE.   PRESS ANY KEY TO BREA
K ORBIT. " :: GOTO 560
550 Y$="YOU HAVE SUCCEEDED I
N SAVING THE MEN!    YOU ARE
 A HERO!!!!!      NOW BREAK
ORBIT BY PRESSING ANY KEY.
 HURRY!!!!!! "
560 Y$=RPT$(" ",27)&Y$ :: FO
R I=1 TO LEN(Y$):: DISPLAY A
T(9,1)BEEP:SEG$(Y$,I,28):: N
EXT I
570 CALL KEY(0,K,S):: IF S=0
 THEN 570 ELSE CALL DELSPRIT
E(#15,#17,#18):: CALL PATTER
N(#16,112,#15,114,#17,112,#1
8,120)
```

```
580 FOR I=6 TO 1 STEP -1 ::
DISPLAY AT(4,1):SEG$("****)
",7-I,1):: CALL SOUND(-999,-
7,0):: NEXT I :: CALL VCHAR(
1,1,32,48)
590 IF F=10 THEN 620 ELSE CA
LL LOAD(-31878,28)
600 FOR I=1 TO 10-F :: CALL
SPRITE(#15+I,104,7,178,20+I*
20):: NEXT I
610 FOR J=1 TO 3 :: FOR I=1
TO 10-F :: CALL PATTERN(#15+
I,100):: NEXT I :: CALL SAY(
"GOODBYE"):: FOR I=1 TO 10-F
 :: CALL PATTERN(#15+I,104):
: NEXT I :: NEXT J
620 FOR I=1 TO 28 :: CALL SO
UND(-999,-5,0):: CALL MOTION
(#I,RND*127-63,RND*127-63)::
 NEXT I
630 FOR I=1 TO 30 :: CALL SO
UND(-4250,-6,I):: NEXT I ::
CALL DELSPRITE(ALL)
640 CALL HCHAR(22,1,32,96)::
 CALL SOUND(1,44733,0):: DIS
PLAY AT(1,1):"PRESS ""ALPHA
LOCK"" DOWN AND PRESS ""A""
FOR ANOTHER METEOR."
650 CALL KEY(0,K,S):: IF S=0
 THEN 650 ELSE IF K<>65 THEN
 RUN "DSK1.LOAD" ELSE MS,SH,
F,HI=0 :: GOTO 230

660 !//MOVE ROCKS//
670 SP=SP+1 :: FOR X=0 TO 13
 STEP 2 :: RANDOMIZE :: Y=DI
(INT(X/2))*(RND*SP+SP-3):: C
ALL MOTION(#X+1,0,Y,#X+2,0,Y
):: NEXT X :: RETURN

680 !///INSTRUCTIONS///
690 PRINT TAB(7);"<<<<<<<<<>
>>>>>>>":TAB(7);"< METEOR RE
SCUE >":TAB(7);"<<<<<<<<<>>>
>>>>>": : : : : : : : : : : :
700 INPUT "NEED INSTRUCTIONS
(Y/N)?":Y$
710 IF Y$<>"Y" THEN 770
720 PRINT : : :" YOU ARE ON
A RESCUE MISSIONTO A METEOR
WHICH IS ON THE VERGE OF ANN
IHILATION.": :" YOUR MISSION
 IS TO RESCUE"
```

```
730 PRINT "THE TEN MINERS ON
 THE METEORWITH THREE LANDER
S.": :" EACH LANDER CAN SUST
AIN    THREE HITS FROM THE F
LYING  ROCKS.": :
740 PRINT " BE CAREFUL! IF Y
OU MANEUVERAND LAND WELL, YO
U'LL BE    SAFE, BUT IF YOU
DON'T, A   HIT COULD BE FATA
L!"
750 PRINT :" AN EXTRA LANDER
 WILL BE    BUILT FOR EACH T
WO MINERS   SAVED. GOOD LUCK
.": : :"PRESS ANY KEY";
760 CALL KEY(0,K,S):: IF S=0
 THEN 760
770 SP=4 :: RETURN

780 !///SUBPROGRAMS///

790 SUB LAND :: CALL LOCATE(
#15,155,117)
800 FOR I=0 TO 30 :: CALL SO
UND(-100,200-I,I):: NEXT I :
: CALL SPRITE(#16,100,7,178,
1)
```

```
810 RANDOMIZE :: I=INT(RND#3
-1):: IF I=0 THEN 810 ELSE C
ALL MOTION(#16,0,I#4)
820 IF I-104 THEN I=100 ELSE
 I=104
830 CALL PATTERN(#16,I):: CA
LL SOUND(-200,-4,15):: CALL
COINC(#16,178,125,2,HIT):: I
F HIT THEN CALL MOTION(#16,0
,0)ELSE 820
840 CALL MOTION(#16,-2,0)::
FOR X=1 TO 20 :: IF I=104 TH
EN I=100 ELSE I=104
850 CALL PATTERN(#16,I):: CA
LL SOUND(80,-5,28):: NEXT X
:: CALL DELSPRITE(#16)
860 SUBEND

870 SUB LS :: CALL LOCATE(#1
5,11,31):: CALL SPRITE(#16,1
00,7,19,35,0,-3)
880 IF I=100 THEN I=104 ELSE
 I=100
```

```
890 CALL PATTERN(#16,I):: CA
LL POSITION(#16,X,Y):: IF Y>
7 THEN 880 ELSE CALL DELSPRI
TE(#16):: CALL SAY("#GOOD WO
RK#")
900 SUBEND

910 SUB BLOW(A,B):: CALL SOU
ND(500,-7,0):: CALL SOUND(30
0,-6,0,110,3):: CALL SOUND(9
00,-5,20,110,20)
920 CALL MOTION(#15,3#(RND#3
+3),RND#5-2)
930 CALL POSITION(#15,X,Y)::
 IF X<169 THEN CALL COINC(AL
L,HIT):: IF HIT THEN CALL SO
UND(60,-5,1):: GOTO 930 ELSE
 930 ELSE CALL MOTION(#15,0,
0)
940 CALL POSITION(#15,X,Y)::
 CALL LOCATE(#15,11,1):: Y=Y
/8+1 :: IF Y>30 THEN Y=30 EL
SE IF Y<2 THEN Y=2
950 CALL VCHAR(23,Y,120):: C
ALL VCHAR(24,Y,121):: CALL V
CHAR(23,Y+1,122):: CALL VCHA
R(24,Y+1,123)
```

```
960 FOR X=0 TO 15 STEP 5 ::
CALL SOUND(X#50+1,-7,X):: NE
XT X :: FOR X=10 TO 30 :: CA
LL SOUND(-100,-7,X):: NEXT X
970 A=A+1 :: IF A>2 THEN SUB
EXIT ELSE IF B=0 THEN 1000 E
LSE CALL SPRITE(#16,104,7,17
8,(Y-1)#8,0,2#SGN((Y-1)#8-12
2))
980 IF I=104 THEN I=100 ELSE
 I=104
990 CALL PATTERN(#16,I):: CA
LL SOUND(-200,-4,15):: CALL
COINC(#16,178,256,3,HIT):: I
F HIT THEN CALL DELSPRITE(#1
6)ELSE 980
1000 CALL MOTION(#15,0,1)
1010 CALL POSITION(#15,X,Y):
: IF Y>29 THEN CALL MOTION(#
15,0,0)ELSE CALL SOUND(-200,
330,9,-3,9):: GOTO 1010
1020 SUBEND
```

```
25000 !NUMTALK, a subprogram
 which allows pronunciation
of numbers correctly in a CA
LL SAY statement
25010 !Can be used in a prog
ram only. Correct format is:
 CALL SAY_NUM(#)
25020 !# Can be any numerica
l data between 0 and 999.
25030 !Keep NUMTALK in MERGE
d format, to be merged with
any program that may be need
ed.
25040 ! Author: Anders Perss
on, Lund, Sweden

25050 SUB SAY_NUM(NR)
25060 IF INITED THEN 25120
25070 DIM TEXT$(33)
25080 RESTORE 25370
25090 FOR I=1 TO 33 :: READ
TEXT$(I):: NEXT I
```

```
25100 NUMPOS$="-+.E012345678
9"
25110 INITED=-1
25120 NUM$=STR$(NR)
25130 IF ABS(NRA)>=1000 OR A
BS(NR)<10 THEN 25210
25140 NEG=(NR<0)
25150 IF NEG THEN NUM$=SEG$(
NUM$,2,LEN(NUM$)):: NR=ABS(N
R):: CALL SAY(TEXT$(1))
25160 IF NR>=100 THEN GOSUB
25240 !SAY HUNDREDS
25170 ON ERROR 25400
25180 IF VAL(NUM$)>=20 THEN
25300 !SAY TY'S
25190 IF VAL(NUM$)>=10 THEN
25350 !SAY TEENS

25200 !SAY DIGITS
25210 FOR I=1 TO LEN(NUM$)::
 CALL SAY(TEXT$(POS(NUMPOS$,
SEG$(NUM$,I,1),1))):: NEXT I
25220 SUBEXIT
```

```
25230 !SAY HUNDREDS
25240 SPEAK$=TEXT$(POS(NUMPO
S$,SEG$(NUM$,1,1),1))&TEXT$(
33)
25250 IF SEG$(NUM$,2,2)<>"00
" THEN SPEAK$=SPEAK$&"+AND"
25260 NUM$=STR$(VAL(SEG$(NUM
$,2,LEN(NUM$)))):: IF NUM$="
0" THEN NUM$=""
25270 CALL SAY(SPEAK$)
25280 RETURN

25290 !SAT TY'S
25300 SPEAK$=TEXT$(VAL(SEG$(
NUM$,1,1))+23)
25310 IF SEG$(NUM$,2,1)<>"0"
 THEN SPEAK$=SPEAK$&"+"&TEXT
$(POS(NUMPOS$,SEG$(NUM$,2,1)
,1))
25320 CALL SAY(SPEAK$):: NUM
$=SEG$(NUM$,3,LEN(NUM$))
25330 GOTO 25210 !TO SAY DIG
ITS
```

```
25340 !SAY TEENS
25350 CALL SAY(TEXT$(INT(VAL
(NUM$))+5)):: NUM$=SEG$(NUM$
,3,LEN(NUM$))
25360 GOTO 25210 !TO SAY DIG
ITS
25370 DATA NEGATIVE,,POINT,4
,ZERO,ONE,TWO,THREE,FOUR,FIV
E,SIX,SEVEN,EIGHT,NINE
25300 DATA TEN,ELEVEN,TWELVE
,THIRTEEN,FOURTEEN,FIFTEEN,S
IX+TEEN,SEVEN+TEEN,EIGHT+TEE
N,NINE+TEEN
25390 DATA TWENTY,THIRTY,FOR
TY,FIFTY,SIXTY,SEVENTY,EIGHT
Y,NINETY,+HUNDRED
25400 RETURN 25410
25410 ON ERROR STOP :: SUBEN
D
```