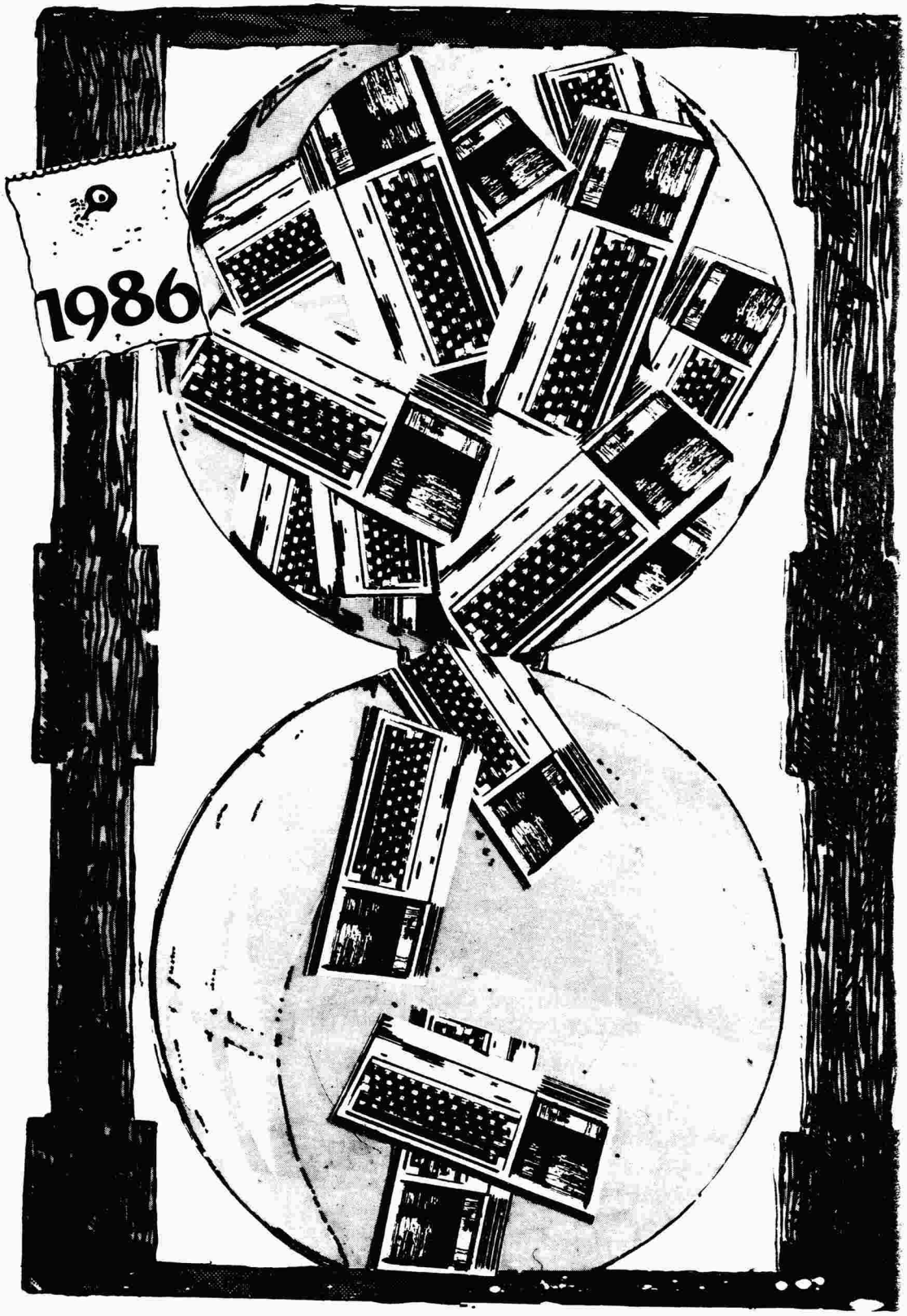
PARCO Electrics 99/4A

INDEPENDENT BI-MONTHLY MAGAZINE FOR THE USERS OF THE T.I. 99/4a HOME COMPUTER



IN THIS ISSUE

*COMPETITION

*Extracts from T.I. Information Package.

*Word Processing

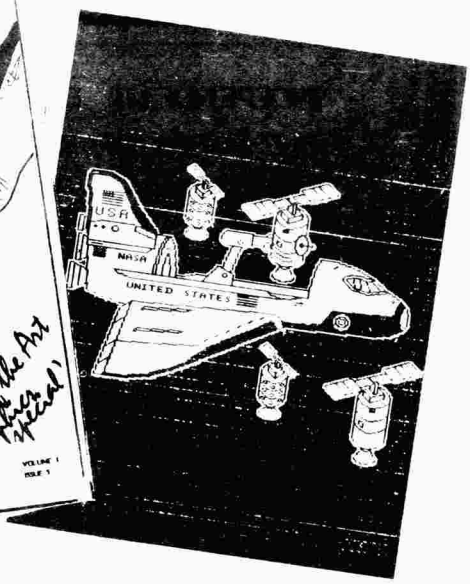
*PROGRAMS

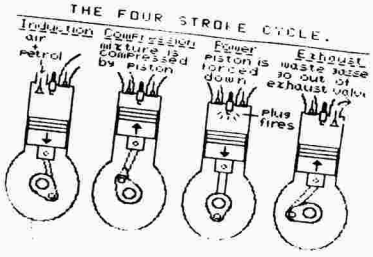
VOLUME 2
ISSUE 1

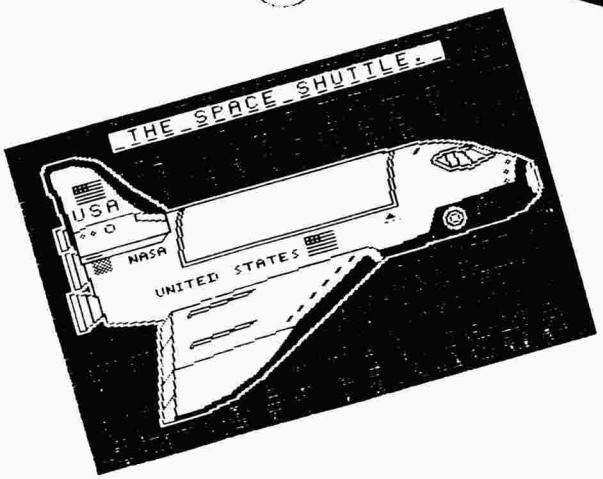
What can GRAPHX do?

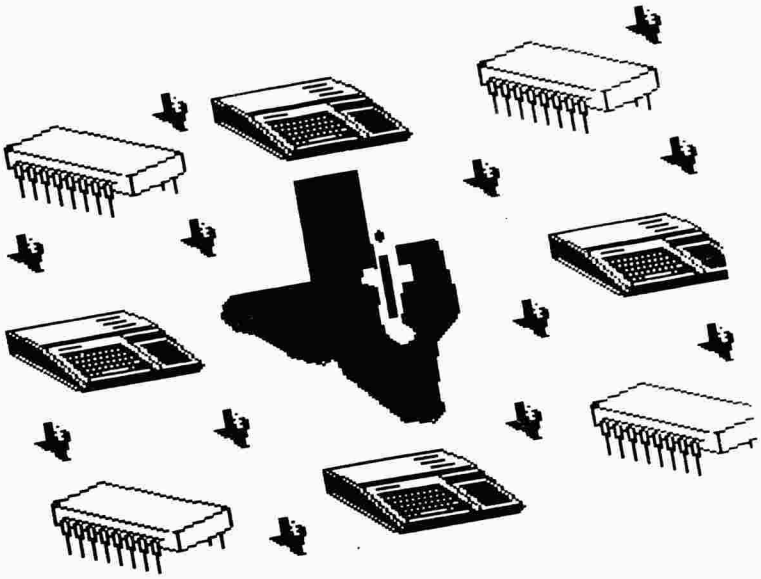
The following illustrations show quite clearly what Graphx is capable of. Obviously these piccies have been reduced to get them onto the page, but just cast a beady over them and see what your TI is capable of with the addition of a few Christmas presents.

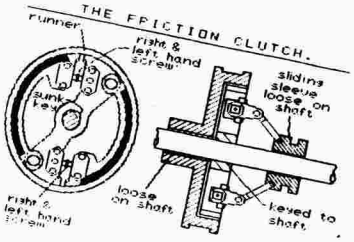
Thanks for these go to, who else, Richard Twyning.

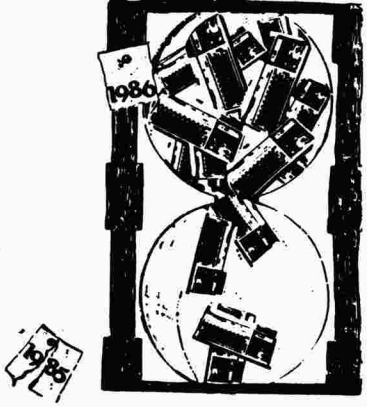












No, I refuse to accept that we have started a second volume. Time simply does not go that fast. Yes, we have had famine, earthquake, volcano, Halley's Comet, Eastenders and other natural phenomena, but not a year of 99/4a, no way. What makes it worse is that due to the delays with which you are now familiar and disproportionately patient, it is more than a year since we started!

Never have we claimed ourselves to be the most comprehensive independent publication ever, but your support indicates all we need to know, and that is that you don't want us to stop yet. Neither do we intend to. So here we go. Not much has changed. It's nice to hope that the standard is improving, but the format is much the same. We won't change unless you want us to, and thus far you seem happy.

The big worry I as editor do have is that the style is possibly a touch too prim and proper for some of you. It has been suggested that a little humour might be allowed to creep in, and even a smattering of irreverence would not go amiss, but honestly, can you imagine such a thing in a noble and upright magazine like this? Answers on a $10 note please, etc.

Spuriously, though, if you are a TI User (and remember that we are here to serve both of you) share what you have. O.K., You may not have earth-shattering programs to submit, and you may not be a smart-alec full of TItbits to dazzle us, but what the heck? There is always someone behind you in terms of experience (me, maybe?) so you DO always have something to share. This is not an elitist mag, so there is no-one ready to sneer at your humble efforts - whether you are young, old, male, female, dumb or smart. Even if you're afraid you haven't got all your facts right, fear not. Far from putting you down, we just leave the door open to anyone who thinks they know better to write in. Anyway, we need people to be asking questions or giving observations and reviews, so get your pens out. The more you chuck in, the more we give out.

Do your bit, and perhaps this time next year I will be introducing volume three.

OCRAP rof - eromdirP yrraH
(I always was a bit backward)

PS. GET THOSE COMPETITION ENTRIES IN AS SOON AS YOU CAN!

CONTENTS:

99/4A MAGAZINE is produced by

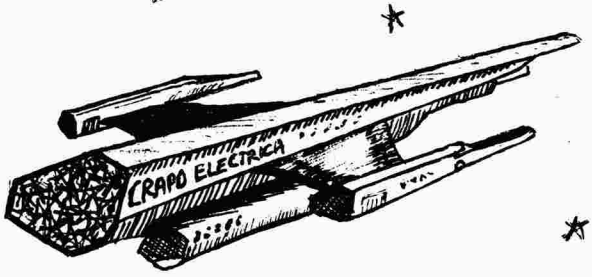
PARCO ELECTRICS
2 Devonshire Court
Heathpark
Honiton
Devon

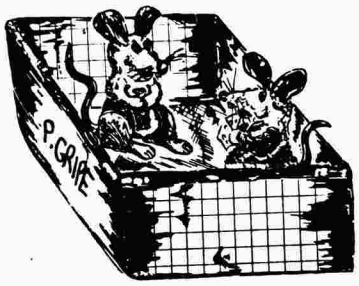
Telephone:

(0404)44425

published by Parco Electrics as a Bi-Monthly Magazine for the users of the **Texas Instruments TI99/4A Home Computer**

price: **£2.25** per issue, annual subscription **£12.00**





THE PARCO STORY CONTINUED....

Life inside Petty Gripe's sandwich box was rather like trying to pull a bird in a Greek Disco. Dark, sweaty, and possibly containing things to be enjoyed if only you could make them out.

You don't need me to tell you that humanity exists solely for the pleasure of mice. Anyone worth his Hitch-hikers will know that we mere mortals are but the subject of their intent scrutiny. Well what a field day Frankly Poorish and his chums had given their rodential majesties.

Their appraisal of the situation acts as a highly illuminating summary of the situation:

Having clambered into Gripe's sandwich box during one of their 'in-the-field' spying missions, the last thing the two mice expected was to be trapped. Humans observed to date had demonstrated an exclusive tendency to devour the entire contents of their luncheon receptacle before snapping the lid shut with a satisfied finality. Not our Gripe. What we have here is a man that finds sufficiency in twelve rounds of sandwich and five cakes. Somehow he never could manage the last three - nobody ever knew why, least of all Rubric his devoted wife, who must spend easily three quarters of her life filling tupperware, and the other quarter packing away Hornby Dublo. There were days when his friends were brave enough to share the cakes, and he always paid them well for their noble efforts, but not today.

The result was, as you will have guessed, the premature closing of said box. Now mice are resourceful creatures. Being confined to such a prison would invoke naked fear and panic in the likes of you and I, but these two mice decided to have a progress meeting.

"Ha Hem," coughed the chairmouse.

"24th December 1985, General Meeting and Progress Report on the section of near-humanity known as the crew of the Starship Crapo Electrica - are you taking notes - as observed on their way to the next mailshot."

"Dossier 1: Frankly Poorish - quiet unassuming mission commander. If this is a soap opera then Frankly is the biggest sud. Recently spending much more time on the bridge of the ship, since relinquishing involuntarily his transgalactic warp visa. Three lightyears of banishment from the interstellar highways for travelling via Gloss-Star at a speed reputed to be faster than sales of Buck Rogers. Not one to sit around and twiddle his extremities though, now even spends some time working. Keeps door locked.

"Dossier 2: Annually Poorish - The first lady of the crew. If this is a soap opera then Annually certainly provides the hotpoint. Not a lady to be messed with, which explains why Gripe has only got nine fingers.

"Dossier 3: Petty Gripe - think of Gripe and the word stalwart comes to mind, whatever it means. If this is a soap opera then Gripe represents carbolics. Gripe unmurmuringly provides backup. Frankly often heard saying " - if anyone gets my backup - " Not given to complaining, in fact the only thing he was ever nearly given to was Oxfam.

"Dossier 4: Rubric Gripe - insists that it is the colourful aspect of the famous cube rather than its shape that lends her the name. If this is a soap opera then Rubric gets the Zanussi award for using the appliance of science to keep her coconut pyramids in shape.

"Dossier 5: Salad Rosebowl - the alluring voice on the phone. If this is a soap opera, then Salad's charms do nothing to deter gents. Recently foiled the aspirations of a thousand male order customers by getting

hitch-hiked to the famous actor Richard Harris, who now poses as a garage mechanic to avoid the trappings of megastardom.

"Dossier 6: Hardly Didmore - all reports say that he fell asleep and nobody noticed.

"Dossier 7: Philament Loving - bright spark extraordinaire, and quite a recent addition to the crew. Not to be confused with Nik Kershaw, who also likes Marmite. Never seen without tool in hand, Phil is the technical one reported to have completely stripped a console to identify I/O error, later established as having been caused by typing OLD CS1 in lower case. If this is a soap opera then Philament is something that got washed up with the Tide. This lad is gifted in ways that you'd never guess. He certainly appears to be keeping everyone here guessing, anyway."

With that, the meeting was suddenly and violently interrupted, as Gripe threw his sandwich box at Poorish.

"Why do you get angry every time I say the word 'Mailshot'?" Frankly shouted, ducking again.

"You can't do that!" yelled Gripe, who had only just ended a period of convalescence as a result of the last mailshot.

The words 'You can't do that' had always been to Frankly the equivalent of a green light. Like the call of a siren to a wayward sailor, 'You can't do that' represented an irresistable challenge to do just whatever it was he was being told he could not do.

"And why not?" He enquired, innocently.

"Look at what the last three have done to me - I'm not a well man." complained Gripe.

Gripe looked frankly quite weak.

Weekly Annually had hinted strongly to Frankly that Gripe was looking weaker daily, and weekly Frankly had replied that if Annually frankly thought that it was strongly more weakly that he was looking daily, then frankly he may need to deal strongly with it on a weekly, if not daily basis.

The last paragraph, however, tired them out, and they all sat down and dozed off for a Kippy's Nightmare or two.

Meanwhile, back at the brunch, in a sandwich box in the corner of the room the meeting was resumed.

"Sundry Reports Of Strange Things That Will Probably Never Be Explained: a summary:-

1) Several persons seen banging on front window of starship to no apparent avail, including one called something like Reg Parrer, shouting:

"33 cases of Blasto at 95p"

another, and not sounding happy, pleading:

"You don't understand, please can you be helping me - the government in my late 2nd cousin's brother's friend's country has fallen, I have had to send my children out to be working in the poppadum mine, and I am still waiting for an SR332k stand-alone parallel RAMdisk control interfacing, inflation is deflating me, my solicitor is dying of terminal earache, what are you going to be doing about it, etc., etc."

2) A group of Richard Twyning impersonators. (Beware of imitations).

3) A bloke clinging for grim life onto a form of one-man girocopter to the rear of the ship and in vain pursuit of it shouting something about thermal paper."

Realizing it was Christmas Eve, Frankly called to Petty, who was frantically loading the sleigh with Modules of Hope."

"Put more on!"

Gripe found it difficult to reply with his head in a box of chips.

"More on!" repeated Poorish.

"Call me names and you can pack this lot yourself!" was the reply.

JOYSTICK EXAMPLE

To move a character around the screen, I can suggest the following programmes as an example:-

```
10  CALL JOYST (1,X,Y)
20  COL = COL + X/4
30  ROW = ROW + Y/4 * -1
40  CALL HCHAR (ROW + 12, COL + 16,42)
50  GOTO 10
```

This will move an asterisk around the screen with joystick control. Should you wish to only see the new position and not the old, simply add other CALL HCHAR lines using character 32 (space), and let it follow the asterisk, erasing as it moves.

Type the example with the Alpha Lock Key DOWN. Run the example with the Alpha Lock UP.

MEMORY SPACE AVAILABLE ROUTINE

There is not command in BASIC to tell how many bytes are left in the memory, however, you may get a good estimate using the following method:-

- After your program is typed in, enter 1 DIM A (1000)
- Then type in RUN

If this statement added in front of your program does not cause a MEMORY FULL error, then you have at least 8030 bytes left. (1 DIM A (1000) takes up approximately 8000 bytes of memory space plus 30 bytes to set up the array).

```
2 DIM B (100) takes up approximately 800 bytes + 30.
3 DIM C ( 10) takes up approximately  80 bytes + 30.
4 DIM C (  1) takes up approximately   8 bytes + 30.
```

The memory space is only allocated after RUN is entered. As you can see, by entering a DIM statement in multiples of 10's, 100's, or 1000's, the largest amount acceptable without a MEMORY FULL error will give a close indication of memory space left.

If "1 DIM A (283)" plus your program will run, but "1 DIM A (284)" plus your program give a MEMORY FULL error, then 283 x 8 = 2264 + 30 = 2294 bytes left.

CHARACTER MOVEMENT USING CALL KEY

```
100 A$ = "183C5AFFFF244281"
110 B$ = "183C5AFFFF242424"
120 CALL CHAR (96,A$)
130 CALL CHAR (97,B$)
140 LET ROW = 12
150 LET COL = 16
160 CALL CLEAR
170 CALL KEY (0,K,S)
180 IF K = 69 THEN 280
190 IF K = 83 THEN 300
200 IF K = 68 THEN 320
210 IF K = 88 THEN 340
220 CALL COLOR (9,2,1)
230 CALL HCHAR (ROW,COL,96)
240 FOR DELAY = 1 TO 100
250 NEXT DELAY
260 CALL HCHAR (ROW,COL, 97)
270 GOTO 160
280 ROW = ROW-1
290 GOTO 230
300 COL = COL-1
310 GOTO 230
320 COL = COL+1
330 GOTO 230
340 ROW = ROW+1
350 GOTO 230
360 END
```

(Defines Character)

(Defines Character)

(Sets 96 = A$)

(Sets 97 = B$)

(Positions Character in centre of the screen

(Check key to see which way it should move)

(Colours character Black)

(Delay loop so that legs are seen to move)

(Changes position of character on screen).

*THE ALPHA LOCK MUST BE DOWN FOR THE PROGRAMMING AND RUNNING THIS EXAMPLE. USE THE UP, DOWN, RIGHT, AND LEFT ARROWS TO MOVE THE CHARACTER AROUND.



TEXAS

PROTECTOR II

You are the last hope! The alien slimehordes of Fraxullus are attacking your cities. One by one, the inhabitants are being exterminated by the invincible Fraxullan mother ship, that is shielded in Baltheric antimatter, making it impossible to destroy. It can however, be paralyzed briefly by laser attacks, but then is able to move more quickly. Be wary of the transporter beam since it spells instant death for your Needlefighter. The Fraxullan mother ship moves relentlessly, drawing up the helpless natives and fiendishly carrying them to Dragonmaw, the sulferous volcano of death. The volcano is due to erupt at any time, but be careful of the explosions that occur occasionally during the game. You must evacuate all 18 people from the City of New Hope before it is too late, and then on to the safety of the Verdann Fortress. You are the sole PROTECTOR!

The Verdann Fortress is located below the horizontal red bar past the vertical laser defense networks that are robot controlled, the Laser Fields of the Straak. The laser installations are stationary while other installations are mobile and track your Needlefighter. The armaments for the Verdann-Fortress are carefully cloaked in invisibility shields until after the eruption of Dragonmaw. At this time the red energy shields are dropped and the Verdann Fortress and it's armaments become available. In order for the survivors from the City of New Hope to reach safety, they must be placed in the escape chute located below the walls of the Fortress. You must save these innocent victims from their gruesome destiny.

You launch your Omicrom Needlefighter from your base deep inside the Xlarr defense post. Refuel your Needlefighter by docking at the refueling pod from where your Needlefighter was originally launched. The alignment of the fighter pod is critical. If it is not exact, you may crash.

In order to transport people from one city to another, you must "hook" each person, one at a time, by passing immediately above the person that you wish to lift. You will know when they are hooked as they will stop waving their arms. After hooking them you must ascend and the person will cling to the bottom of your Needlefighter. In order to "drop" a person you merely fly over the area where you wish to deposit the person (making sure that they touch the disembarking area) and he/she will pop off.

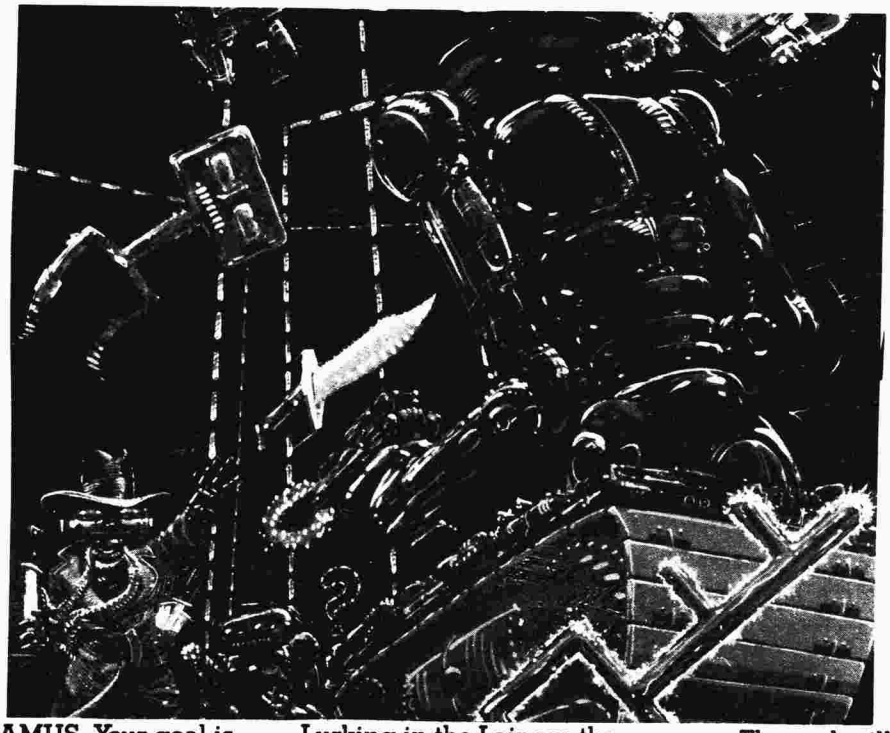
All of the people must first be brought to the City of New Hope before the barrier to the Verdann Fortress will be opened. It is possible (in some of the easier levels) to catch people before they sink into the mouth of Dragonmaw. This maneuver requires great skill and timing. Also the unpredictability of the volcanic eruptions makes this a very risky procedure.

The aliens and fate conspire against you at every turn. The Fraxullan mother ship launches Xytonic PULSE-TRACKERS to destroy you. These fiendish trackers pursue you across the skies. They are persistent and will destroy your passengers (and/or the Needlefighter) if they make contact with them. Be wary of the CHOMPERS, an indigenous life form. They are powerful and unpredictable. In fact, they eat Needlefighters for breakfast! Also, there are rocket bases all over the surface of the planet. These rockets fire in many directions. You may destroy rocket installations by hitting them with laser cannon fire when they turn red.

Painstakingly, after every man, woman and child have been flown past Dragonmaw and placed in the City of New Hope, the volcano will erupt, starting a lava flow that inevitably destroys the City of New Hope. It is here that you find out if you passed the ultimate test. Your mouth is dry, and the sweat trickles down your face, as you strain against the impossible odds. There is no time to think, only to do! It has fallen on your shoulders, you are the last hope. You are the PROTECTOR!

SHAMUS

REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW
REVIEW



TELL THE SHADOW THAT SHAMUS IS HERE!

You are the SHAMUS. Your goal is to reach the very core of the SHADOW's Lair and destroy him. This is accomplished by progressively exploring each level of the Lair and accumulating the greatest number of points, bonuses, and extra lives. The Lair consists of four levels, the colors in order of progressive difficulty are BLACK, BLUE, GREEN, and RED. You gain extra points for completely clearing each room of intruders. In the Lair you will encounter POD rooms, which exist in another dimension and have only a small time window which must be opened to gain entrance.

Each of the colored levels of the Lair contains various colored Keys which are paired with matching colored Keyholes. These Keys are picked up by SHAMUS by touching them. As they are touched they are displayed in the right-hand bottom of the screen. When encountering a Keyhole of the matching color, SHAMUS merely makes contact with the Keyhole and the wall retracts permitting you to proceed through the Lair. If the wall does not drop you have not obtained the correct color coded Key, and you will not be permitted past the Keyhole. This is the only way to gain entry from one color level to another.

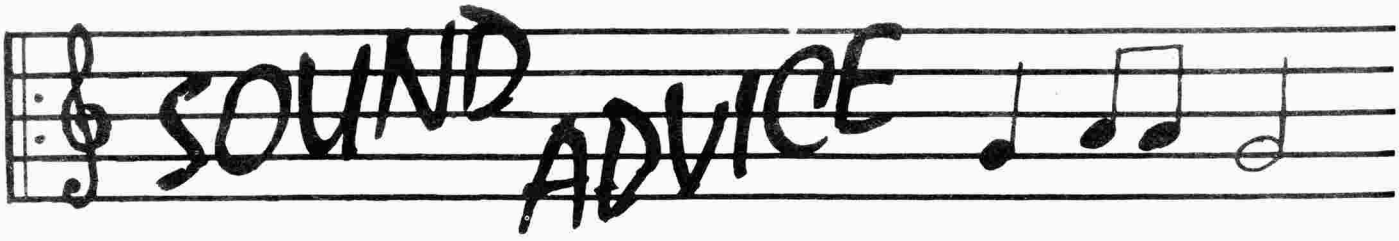
Lurking in the Lair are the SHADOW's dangerous henchmen to overcome. Never underestimate the ferocity of the SPIRAL DRONES. They are sneaky, and intensely sensitive to movement within the passages of the Lair. They are armed with disruptors. Relatively slow moving semi-mechanical androids are the ROBO-DROIDS. These droids are methodical and never give up. They are also armed with disruptors. SNAP JUMPERS are self-propelled annihilators who cannot think. This feature, however, makes them all the more dangerous. They are a strange mutation who exist simultaneously, in two dimensions. They "snap" in and out of your time-space continuum and are therefore unpredictable and because of this quality are extremely quick. Their speed makes them very dangerous. SHAMUS' only weapons are the ION-SHIVS (Ionic-Short High Intensity Vaporizers). These weapons are banned in every part of the galaxy. They totally disintegrate any life form upon contact, unless shielded by Tri-Gamma body armor. There can only be two ION-SHIVS on the screen at one time.

The arch villain himself, the SHADOW is constantly monitoring your activities in his Lair. Be extremely careful when you hear a low pitched pulsing begin. This is a signal that the SHADOW is about to enter the room. You cannot kill the SHADOW since he wears Tri-Gamma armor. You can, however, stun him. He remains stunned for a short time, after which you had better move quickly!

The current number of lives are displayed in the upper right-hand corner of the screen. If you encounter a bubbling bottle in a room you need merely to touch it to obtain an extra life. By passing over pulsating question marks you will invite either bonus points or an extra life.

Unlike other text adventures, SHAMUS can only be mastered by a long and arduous training period, in which your reflexes are sharpened to a point where you can deal with the incredible speed and viciousness of the attacks of the Shadow and his henchmen. Only in the deepest recesses of level RED will you find the answer to this puzzle! Tell the SHADOW that SHAMUS is here!

2021 clarification: SHAMUS is NOT a text adventure. It is a graphical maze / shoot em up on module.



Let's tune in with a contribution from Phil 'Fingers' Donald.

Dear Sirs,

I'm no authority on music, even though I have been playing guitar for about ten years now (ask my neighbours) and I have also been known to tickle the ivories from time to time, but the one thing that was lacking in my repertoire (I think that's how you spell it) was the ability, or lack of it, to read a note from a bass or treble clef.

But just recently I purchased 'Music Maker' from Parco, and to be honest I have come on in leaps and bounds, and music isn't such a daunting task after all. In fact I would say it's simple.

There are only two drawbacks with Music Maker. One is that it sounds like an organ (the keyboard kind) and the other is that you can't change counts in the middle of a piece - but you seldom have to.

I don't know the current price of Music Maker now, but whatever it is - BUY IT!!!

If anybody out there needs a song 'sussing out' send me a cassette with the song on it and an SAE and I'll send you the computer version. No sissy music please - i.e. Boy George, Nik Kershaw, or any of those schoolgirl wimps. Rock only.

Yours,

Phil Donald

Phil Donald
16 Chalford Road
Newall Green
Wythenshawe
Manchester
M23 8SG

Here is a program what Phil wrote called CHORDS which plays all the major chords from A to G sharp. The program also displays all the notes on bass and treble clefs.

```
100 REM CHORDS
110 REM BY PHIL DONALD
120 REM FOR PARCO MAGAZINE
130 CALL CLEAR :: PRINT "DEF
INING CHARS NOW" :: CALL MAG
NIFY(3):: CALL CHAR(116,"010
10101010101011D3F7F7F3E1C"&R
PT$("0",36))
140 CALL SCREEN(16):: RESTOR
E 150 :: FOR LC=1 TO 41 :: R
EAD CH,CH$ :: CALL CHAR(CH,C
H$):: NEXT LC
150 DATA 33,FF,34,0000010102
020404,35,0808081010101020,3
6,2020204040404040,37,404040
202010100,38,020100000000000
0
160 DATA 39,0106040404040300
,42,0000000101010101,43,0202
020202020202,44,030306060A12
2242,58,4282020202020202
170 DATA 59,02020202031E2222
,60,4242424242424262,61,1E02
020202020202,62,02FF01010101
0101,63,01010101010101FE
180 DATA 64,000000001E214040
,91,8080800000000000,92,0001
0106081060C0,93,800000000000
0000,94,00000000FC030000
190 DATA 95,0000000000010204
,96,18E0000000000000,97,0000
000000008040,98,4040404040408080,99,000000000000C020
200 DATA 100,1010101010101020,101,2020404080000000,102,0
0000000030C1020,103,40808080
80808080,104,808047671F000000
0
210 DATA 105,0618608000000000,106,0000003FC0000000,107,0
0000000030C106080,108,000000C0
700C0601,109,010000000000000
0
220 DATA 110,0000000000000101,111,0102020204081020,112,2
040800000000000.113,8080C040
40404040,...4,4040808080800000
0
230 CALL CLEAR :: CALL CHAR(
120,"2424FF2424FF2424"&RPT$(
"0",48),124,"4040405E62424C7
0"&RPT$("0",48))
240 CALL CHAR(128,"010101010
10101011D3F7FFF7F3E1C"&RPT$(
"0",36)):: CALL COLOR(14,9,9
)
250 CALL HCHAR(1,1,136,32)::
 CALL HCHAR(24,1,136,32):: C
ALL VCHAR(1,2,136,24):: CALL
 VCHAR(1,32,136,24):: DISPLA
Y AT(4,13)SIZE(5):"MUSIC"
260 DISPLAY AT(6,14)SIZE(2):
"BY" :: DISPLAY AT(8,10)SIZE
(11):"PHIL DONALD" :: DISPLA
Y AT(11,4)SIZE(22):"A TEACH
YOURSELF MUSIC"
270 DISPLAY AT(13,4)SIZE(21)
:"GUIDE TO BOTH READING" ::
DISPLAY AT(15,4)SIZE(18):"AN
D WRITING MUSIC."
280 DISPLAY AT(22,2)SIZE(26)
:"  PRESS ANY KEY FOR MENU
"
290 CALL KEY(0,K,S):: IF S=0
 THEN 290
300 CALL CLEAR :: DISPLAY AT
(5,14):"MENU" :: DISPLAY AT(
6,14):"----" :: DISPLAY AT(1
0,2):"PRESS"
310 CALL COLOR(14,5,5):: CAL
L HCHAR(1,1,136,32):: CALL H
CHAR(24,1,136,32):: CALL VCH
AR(1,2,136,24):: CALL VCHAR(
1,32,136,24)
320 DISPLAY AT(14,2):"1)  CH
ORDS" :: DISPLAY AT(16,2):"2
)  TREBLE AND BASS CLEF" ::
DISPLAY AT(18,2):"3)  END"
330 CALL KEY(0,K,S):: IF S=0
 THEN 330 :: K=K-48 :: IF K<
1 OR K>3 THEN 330 ELSE ON K
GOTO 710,360,340
340 CALL CLEAR :: CALL SOUND
(500,294,1,440,1,587,1):: CA
LL SOUND(400,220,1,330,1,440
,1):: CALL SOUND(1000,294,1,
440,1,587,1)
350 END
360 CALL SCREEN(8):: CALL CL
EAR :: FOR ROW=2 TO 10 STEP
2 :: CALL HCHAR(ROW,6,33,26)
:: NEXT ROW :: FOR ROW=14 TO
 22 STEP 2
370 CALL HCHAR(ROW,6,33,26):
: NEXT ROW :: RESTORE 380 ::
 FOR CH=1 TO 41 :: READ RX,C
X,DX :: CALL HCHAR(RX,CX,DX)
:: NEXT CH
380 DATA 5,2,34,6,2,35,7,2,3
6,8,2,37,9,2,38,10,2,39,2,3,
42,3,3,43,4,3,44,5,3,58,6,3,
59,7,3,60
390 DATA 8,3,61,9,3,62,10,3,
63,1,4,64,2,4,91,3,4,92,4,4,
93,6,4,94,8,4,95,9,4,96,1,5,
97,2,5,98
```

```
400 DATA 3,5,93,6,5,99,7,5,1
00,8,5,101,14,2,102,15,2,103
,16,2,104,19,2,105,14,3,106,
18,3,107
410 DATA 14,4,108,15,4,109,1
6,4,110,17,4,111,18,4,112,15
,5,113,16,5,114
420 RESTORE 430 :: FOR SP=1
TO 21 :: READ LX,LY :: CALL
SPRITE(#SP,116,2,LX,LY):: NE
XT SP
430 DATA 158,41,151,61,143,8
1,134,101,126,121,119,141,11
1,161,102,181,94,201,89,221
440 DATA 78,41,68,61,63,81,5
5,101,46,121,38,141,31,161,2
3,181,14,201,5,221,255,241
450 FOR NO=1 TO 21 :: READ L
X,LY,LC :: CALL HCHAR(LX,LY,
LC):: NEXT NO
460 DATA 23,6,71,22,9,65,21,
11,66,20,14,67,19,16,68,18,1
9,69,17,21,70,16,24,71,15,26
,65,14,29,66
470 DATA 13,6,67,12,9,68,11,
11,69,10,14,70,9,16,71,8,19,
65,7,21,66,6,24,67,5,26,68,4
,29,69,3,31,70
480 FOR R=1 TO 3 :: CALL COL
OR(#1,R+11):: CALL SOUND(500
,196,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#1,5)
490 FOR R=1 TO 3 :: CALL COL
OR(#2,R+11):: CALL SOUND(500
,220,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#2,5)
500 FOR R=1 TO 3 :: CALL COL
OR(#3,R+11):: CALL SOUND(500
,247,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#3,5)
510 FOR R=1 TO 3 :: CALL COL
OR(#4,R+11):: CALL SOUND(500
,262,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#4,5)
520 FOR R=1 TO 3 :: CALL COL
OR(#5,R+11):: CALL SOUND(500
,294,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#5,5)
530 FOR R=1 TO 3 :: CALL COL
OR(#6,R+11):: CALL SOUND(500
,330,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#6,5)
540 FOR R=1 TO 3 :: CALL COL
OR(#7,R+11):: CALL SOUND(500
,349,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#7,5)
550 FOR R=1 TO 3 :: CALL COL
OR(#8,R+11):: CALL SOUND(500
,392,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#8,5)
560 FOR R=1 TO 3 :: CALL COL
OR(#9,R+11):: CALL SOUND(500
,440,1):: GOSUB 1170 :: NEXT
 R :: CALL COLOR(#9,5)
570 FOR R=1 TO 3 :: CALL COL
OR(#10,R+11):: CALL SOUND(50
0,494,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#10,5)
580 FOR R=1 TO 3 :: CALL COL
OR(#11,R+11):: CALL SOUND(50
0,523,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#11,5)
590 FOR R=1 TO 3 :: CALL COL
OR(#12,R+11):: CALL SOUND(50
0,587,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#12,5)
600 FOR R=1 TO 3 :: CALL COL
OR(#13,R+11):: CALL SOUND(50
0,659,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#13,5)
610 FOR R=1 TO 3 :: CALL COL
OR(#14,R+11):: CALL SOUND(50
0,698,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#14,5)
620 FOR R=1 TO 3 :: CALL COL
OR(#15,R+11):: CALL SOUND(50
0,784,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#15,5)
630 FOR R=1 TO 3 :: CALL COL
OR(#16,R+11):: CALL SOUND(50
0,880,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#16,5)
640 FOR R=1 TO 3 :: CALL COL
OR(#17,R+11):: CALL SOUND(50
0,988,1):: GOSUB 1170 :: NEX
T R :: CALL COLOR(#17,5)
650 FOR R=1 TO 3 :: CALL COL
OR(#18,R+11):: CALL SOUND(50
0,1047,1):: GOSUB 1170 :: NE
XT R :: CALL COLOR(#18,5)
660 FOR R=1 TO 3 :: CALL COL
OR(#19,R+11):: CALL SOUND(50
0,1175,1):: GOSUB 1170 :: NE
XT R :: CALL COLOR(#19,5)
670 FOR R=1 TO 3 :: CALL COL
OR(#20,R+11):: CALL SOUND(50
0,1319,1):: GOSUB 1170 :: NE
XT R :: CALL COLOR(#20,5)
680 FOR R=1 TO 3 :: CALL COL
OR(#21,R+11):: CALL SOUND(50
0,1397,1):: GOSUB 1170 :: NE
XT R :: CALL COLOR(#21,5)
690 DISPLAY AT(24,1):"Y TO R
EPLAY OR N FOR MENU"
700 CALL KEY(0,K,S):: IF S=0
 THEN 700 :: IF K=89 THEN CA
LL DELSPRITE(ALL):: GOTO 420
 ELSE IF K<>78 THEN 700 :: C
ALL DELSPRITE(ALL):: GOTO 30
0
710 A=110 :: B=123 :: C=131
:: D=147 :: E=165 :: F=175 :
: G=196 :: AA=220 :: BB=247
:: CC=262 :: DD=294 :: EE=33
0 :: FF=349
720 GG=392 :: AAA=440 :: BBB
=494 :: CCC=523 :: DDD=587 :
: EEE=659 :: FFF=698 :: GGG=
784 :: CALL SCREEN(16)
730 CALL CLEAR
740 PRINT :"CHORD
   PRESS": : :"A
      A":"B FLAT/A SHARP
   BF":"B
 B":"C                    C"
750 PRINT :"C SHARP/D FLAT
   CS":"D
 D":"E FLAT/D SHARP      EF"
:"E                   E":"F
                    F"
760 PRINT :"F SHARP/G FLAT
   FS":"G
 G":"G SHARP/A FLAT      GS"
: :"Q TO QUIT": :
770 INPUT "SELECT PLEASE ":N
$ :: IF N$="Q" THEN 300 :: C
ALL CLEAR
780 FOR ROW=2 TO 10 STEP 2 :
: CALL HCHAR(ROW,6,33,26)::
NEXT ROW :: FOR ROW=14 TO 22
 STEP 2 :: CALL HCHAR(ROW,6,
33,26):: NEXT ROW
790 RESTORE 380 :: FOR CH=1
TO 41 :: READ RO,CO,DO :: CA
LL HCHAR(RO,CO,DO):: NEXT CH
800 IF N$="A" THEN 810 ELSE
830
810 CALL SPRITE(#1,116,2,94,
80,#2,116,2,63,80,#3,116,2,3
8,80):: CALL SOUND(1000,AA,1
,EE,1,AAA,1)
820 DISPLAY AT(23,1):"A MAJO
R CHORD" :: GOSUB 1180 :: GO
TO 810
830 IF N$="BF" THEN 840 ELSE
 860
840 CALL SPRITE(#1,116,2,89,
80,#2,116,2,55,80,#3,116,2,3
1,80,#4,124,3,95,68,#5,124,3
,36,68)
850 CALL SOUND(1000,233,1,FF
,1,466,1):: DISPLAY AT(23,1)
:"B FLAT MAJOR CHORD" :: GOS
UB 1180 :: GOTO 840
860 IF N$="B" THEN 870 ELSE
890
870 CALL SPRITE(#1,116,2,89,
80,#2,116,2,55,80,#3,116,2,3
1,80,#4,120,5,62,40):: DISPL
AY AT(23,1):"B MAJOR CHORD"
880 CALL SOUND(1000,BB,1,370
,1,BBB,1):: GOSUB 1180 :: GO
TO 870
890 IF N$="C" THEN 900 ELSE
920
900 CALL SPRITE(#1,128,2,78,
80,#2,116,2,63,80,#3,116,2,4
6,80):: DISPLAY AT(23,1):"C
MAJOR CHORD"
910 CALL SOUND(1000,CC,1,EE,
1,66,1):: GOSUB 1180 :: GOTO
 900
920 IF N$="CS" THEN 930 ELSE
 950
930 CALL SPRITE(#1,128,2,78,
80,#2,116,2,46,80,#3,116,2,2
3,80,#4,120,5,86,40,#5,120,5
,54,40)
940 CALL SPRITE(#6,120,5,30,
40):: DISPLAY AT(23,1):"C SH
ARP MAJOR CHORD" :: CALL SOU
ND(1000,277,1,415,1,554,1)::
 GOSUB 1180 :: GOTO 930
950 IF N$="D" THEN 960 ELSE
980
960 CALL SPRITE(#1,116,2,68,
80,#2,116,2,38,80,#3,116,2,1
4,80):: CALL SOUND(1000,DD,1
,AAA,1,DDD,1)
970 DISPLAY AT(23,1):"D MAJO
R CHORD" :: GOSUB 1180 :: GO
TO 960
980 IF N$="EF" THEN 990 ELSE
 1020
990 CALL SPRITE(#1,116,2,63,
80,#2,116,2,31,80,#3,116,2,5
,80,#4,124,3,68,68,#5,124,3,
36,68)
1000 CALL SPRITE(#6,124,3,13
,68):: DISPLAY AT(23,1):"E F
LAT MAJOR CHORD" :: CALL SOU
ND(1000,311,1,466,1,622,1)
1010 GOSUB 1180 :: GOTO 990
1020 IF N$="E" THEN 1030 ELS
E 1050
1030 CALL SPRITE(#1,116,2,63
,80,#2,116,2,31,80,#3,116,2,
5,80):: DISPLAY AT(23,1):"E
MAJOR CHORD"
1040 CALL SOUND(1000,EE,1,BB
,1,EEE,1):: GOSUB 1180 :: GO
TO 1030
1050 IF N$="F" THEN 1060 ELS
E 1080
1060 CALL SPRITE(#1,116,2,55
,80,#2,116,2,23,80,#3,116,2,
255,80):: DISPLAY AT(23,1):"
F MAJOR CHORD"
1070 CALL SOUND(1000,FF,1,CC
,1,FFF,1):: GOSUB 1180 :: GO
TO 1060
1080 IF N$="FS" THEN 1090 EL
SE 1110
1090 CALL SPRITE(#1,116,2,55
,80,#2,116,2,23,80,#3,116,2,
255,80,#4,120,5,62,40,#5,120
,5,30,40)
1100 CALL SPRITE(#6,120,5,5,
40):: DISPLAY AT(23,1):"F SH
ARP MAJOR CHORD" :: CALL SOU
ND(1000,370,1,554,1,740,1)::
 GOSUB 1180 :: GOTO 1090
1110 IF N$="G" THEN 1120 ELS
E 1140
1120 CALL SPRITE(#1,116,2,10
2,80,#2,116,2,68,80,#3,116,2
,46,80):: DISPLAY AT(23,1):"
G MAJOR CHORD"
1130 CALL SOUND(1000,66,1,DD
```

```
,1,666,1):: GOSUB 1180 :: GO
TO 1120
1140 IF N$="GS" THEN 1150 EL
SE 730
1150 CALL SPRITE(#1,116,2,10
2,80,#2,116,2,63,80,#3,116,2
,46,80,#4,120,5,109,40,#5,12
4,3,68,68)
1160 CALL SPRITE(#6,120,5,54
,40):: DISPLAY AT(23,1):"G S
HARP MAJOR CHORD" :: CALL SO
UND(1000,415,1,622,1,831,1):
: GOSUB 1180 :: GOTO 1150
1170 FOR DEL=1 TO 150 :: NEX
T DEL :: RETURN
1180 DISPLAY AT(24,1):"PLAY
CHORD AGAIN Y/N"
1190 CALL KEY(0,K,S):: IF S=
0 THEN 1190 :: IF K=89 THEN
CALL DELSPRITE(ALL):: RETURN
 ELSE 1200
1200 IF K<>78 THEN 1190 :: C
ALL DELSPRITE(ALL):: GOTO 73
0
```

Thanks, fingers.

Now a review of a book that, for once, goes into some depth into sound and speech on the TI. It's one of the excellent 'COMPUTE!' series, and is called 'TI99/4a SOUND and GRAPHICS'.

Half of the book covers graphics, as you may conclude from the title. It's really geared toward EB owners, as it moves from standard graphics techniques to the exciting world of advanced sprite-handling. The other half of the book is what interests us here though.

What guarantees this book a place on my shelf is the fact that MUSIC is catered for rather than just SOUND. I know that Iron Maiden fans didn't know that there was a difference, but in fact there is. Yes the book does show you how to create the sounds of Rocket, Morse Code, Computer(?), Sirens, Bombs, Bells and Explosions, but it is a fleeting look compared to the attention given to music. Using sample programs and charts, the reader is given considerable insight into the relationship between music as written down and music as programmed on the TI. Put it this way - how many books have you got for the TI that go to such lengths that the the terms Andantino, Prestissimo and Diminuendo get a mention?

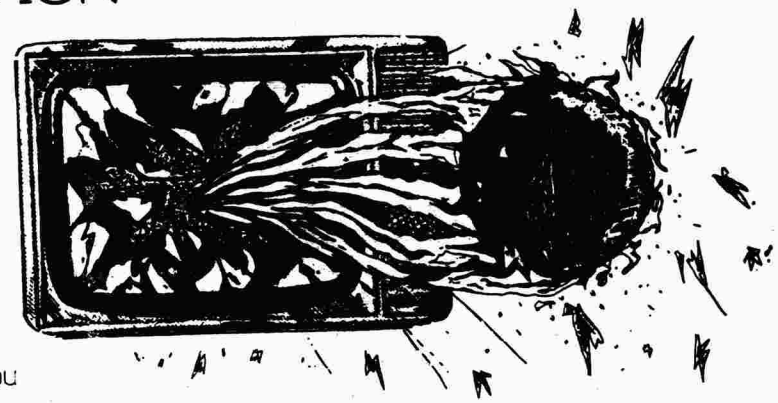
Then speech - and again more charts and sample programs to help you get the most from your synth. Unusually we find a book that offers help to those who have been lucky enough to invest in 'Text-to-Speech', but who don't know quite what to do with it!

The fact that this book was worth mentioning in the SOUND ADVICE column indicates what we think of it.

Don't forget to follow Phil Donald's example and send in a SOUND proposition of your own,

SOUNDing out now till next time.

HALLEYS COMPETITION



The last competition was so well recieved, that we simply have to run another.

This time we thought it would be topical and fun to present you with the challenge of writing a program to be called 'HALLEYS COMET'.

The program can take any form you like - Graphics, Text, Adventure, even Music; and can be in Basic, Extended Basic or Assembly Language. The limitations of the language you use will be taken into consideration, so the winner will be judged on its own merit in that context. One thing we do ask is that it be your own work. Sadly one or two entrants in the last competition pinched other programs and modified them. 'Nuff said.

What's at the end of it for the winner? A FIFTY QUID VOUCHER that's what! Fifty note's worth to spend at PARCO - definitely good news, eh?

CLOSING DATE IS **28th FEB 1986** SO YOU HAVE OVER A MONTH TO WORK ON YOUR PROGRAM. RECORD YOUR ENTRY ON TAPE OR DISK PLEASE WITH AN SAE IF YOU WANT IT BACK!

Get on with it then.

KEYBOARD KRUNCHERS

The only problem with setting a high standard is that it has to be maintained! We have made life hard for ourselves by bringing you lots of great krunchables in Vol.1, and now we are into Vol.2 we can´t afford to let things slip.

Thankfully you are still sending good material in, so our job is made a little easier. In actual fact we still enjoy something of a backlog of entries, and some of the following programs were entries in our popular competition of Vol.1.

´Beat the Robots´ in Extended Basic is excellent, as you will discover if you take the trouble to type it in. Really it is several games in one, since it has four distinct stages, each a game in its own right.

As usual, we are glad of your comments, especially if they are complimentary. Naturally we feel it is the contributors that deserve the praise, and if you have enjoyed a particular program why not write in and tell us, so that the author can get a little big-headed for a day or two?

´Loan Calculations´ is pretty self-explanatory. Many have requested Home/Office utilities, and we hope that many of you will benefit from using this program in TI Basic.

Bruce Forsyth doesn´t actually make an appearance in ´Turn of the Card´, but a certain TV game was the inspiration for this TI Basic effort. This is one of those games that´s hard to stop, once you´ve started playing, you have been warned. This is where you lose all the money you´ve borrowed with Loan Calculations.

Talking of cards, our Bright Spark Graham Marshall has turned up trumps with a very effective scrolling routine that comes in three versions, for those of you with expanded systems and technical dispositions. We know you will be impressed with this, and it satisfies the cry of many for more Assembly Language routines in ´99/4a´.

KRUNCH * KRUNCH

KEYBOARD KRUNCHERS
99/4a Magazine
Parco Electrics
2 Devonshire Court
Heathpark
Honiton
Devon

****
* BEAT THE ROBOTS ********** J.Milsom *
****
Extended Basic

So, 007, you think you can BEAT THE ROBOTS eh? Well let me tell you that it ain´t that simple. First of all you have to figure out the combination of the safe that contains the key to the reactor that is in the nuclear power station that has been taken by Robots! If you manage that, then you have to continue your race against time by negotiating the forest that lies between you and the station. Assuming that you achieve this, then there are the robots to overcome. If you haven´t already run out of time by this stage, then the last desperate challenge awaits you, for you must wind your way up through the passages to the reactor itself. Phew!

This program is written in Extended Basic, but the smart ones among you will see that conversion to TI Basic would not be too hard.

PLEASE NOTE THAT CONTROL CHARACTERS ARE USED IN THIS PROGRAM. AFTER TYPING IN LINE 280, ´RUN´ THE PROGRAM TO INITIALIZE THE GRAPHICS CHARACTERS DEFINED DURING THE EARLY PART. THEN, WHEN YOU TYPE IN LINE 290, HOLD ´CTRL´ DOWN WHILE PRESSING THE UPPER-CASE CHARACTERS OF THAT LINE. LIKEWISE LINES 1430, 1440, AND ALSO THE STRING OF ´C´s IN LINE 1680. THANKYOU.

Beat the Robots

```
100 OPTION BASE 1
110 REM CRACK SAFE
120 DIM H$(40)
130 REM DEF CHAR'S AND TITLE
140 RANDOMIZE :: CALL CLEAR
150 CALL CHAR(123,"183C183C5
A182442")
160 CALL CHAR(129,"FF8080808
0808080"):: CALL CHAR(130,"F
C0202010101102FC"):: CALL CHA
R(131,"FF")
170 CALL CHAR(132,"0F1020408
0808080F0808040201010101808080
808080808080101010101010101"
)
180 CALL CHAR(136,"FF8080808
08080FF0000000000000FFFF010
1010101013F4080808080403F"
)
190 CALL CHAR(140,"804020100
80402018080808040201 00F01010
10102040 8F0"):: CALL CHARPAT
(46,C$):: CALL CHAR(120,C$)
200 CALL CHAR(124,"040705070
603010F1717161606060 61C20E0A
0E060C080F0E8E8686860606038"
)
210 CALL CHAR(96,"FFE7E7C3C3
E7FFFF183C7E183C7E1800"):: C
ALL CHAR(99,"FFFFFFFFFFFFFF
FF")
220 CALL CHAR(106,"80C0E0F0F
8FCFEFF0103070FF1F3F7FF"):: 
CALL CHAR(104,"3C7E7E7E3C181
800"):: CALL CHAR(112,"00007
84CFFFF6600FFFFDDFFFFDDFFFF"
)
230 CALL CHAR(95,"2040906048
302458"):: CALL CHAR(102,"00
0000FFFF"):: CALL CHAR(103,"
55AA55AA55AA55AA"):: CALL CH
AR(105,"18185A5A7E7E6666")
240 CALL COLOR(9,2,5,10,2,14
,11,2,7,12,2,12,13,5,1,14,5,
1)
250 CALL MAGNIFY(4):: CALL T
ITLE
260 REM CRACK SAFE
270 T=60 :: GAME=1 :: CALL Q
(GAME,T)
280 CALL CHAR(132,"808080808
08080FF0101010101010 1FF")
290 DISPLAY AT(4,9)ERASE ALL
:"F          G" :: DISPLAY
AT(5,9):"F  b a r y  G" :: D
ISPLAY AT(6,9):"DIIIIIIIIIII
E"
300 CALL SCREEN(4)
310 DISPLAY AT(1,1):"TIME LE
FT = ";T;"YOU MUST PRESS THE
SE BUTTONS    IN THE CORRECT
 ORDER"
320 DISPLAY AT(8,10):"GUESS
 NO.CORRECT"
330 A$="bryn" :: RW=9
340 A=INT(RND*4)+1
350 B=INT(RND*4)+1 :: IF B=A
 THEN 350
360 C=INT(RND*4)+1 :: IF C=A
 OR C=B THEN 360
370 D=INT(RND*4)+1 :: IF D=A
 OR D=B OR D=C THEN 370
380 B$=SEG$(A$,A,1)&SEG$(A$,
B,1)&SEG$(A$,C,1)&SEG$(A$,D,
1)
390 R=0 :: RW=RW+1 :: T=T-1
:: IF RW=19 THEN RW=10
400 CALL KEY(5,K,S)
410 ACCEPT AT(RW,10)SIZE(4)B
EEP VALIDATE("bary"):G$
420 DISPLAY AT(1,1):"TIME LE
FT = ";T :: IF T<=0 THEN 134
0
430 FOR I=1 TO 4 :: IF SEG$(
G$,I,1)=SEG$(B$,I,1)THEN R=R
+1
440 NEXT I
450 DISPLAY AT(RW,20):R :: I
F R=4 THEN DISPLAY AT(20,1):
"YOU HAVE CRACKED THE SAFE
  AND NOW HAVE THE KEY" :: G
OTO 470
460 IF R<4 THEN DISPLAY AT(2
0,10):"TRY AGAIN" :: GOTO 39
0
470 CALL W(900)
480 REM FOREST RIDE GAME 2
490 GAME=2 :: CALL Q(GAME,T)
500 CALL CLEAR :: CALL SCREE
N(8):: CALL COLOR(9,13,1,10,
3,1,11,7,1)
510 FOR I=1 TO 40
520 READ H$(I)
530 NEXT I
540 CO=19 :: CALL HCHAR(13,C
O,112)
550 FOR I=1 TO 40
560 CALL HCHAR(13,CO,32):: P
RINT H$(I):: IF CO<24 THEN T
=T-.5 ELSE T=T-1
570 CALL KEY(3,K,S):: CALL J
OYST(2,RO,COL):: IF K=83 OR
RO=-4 THEN 580 ELSE IF K=68
OR RO=4 THEN 590 ELSE 600
580 CO=CO-1 :: GOTO 600
590 CO=CO+1
600 CALL GCHAR(13,CO,6):: CA
LL HCHAR(13,CO,112):: IF G=3
2 THEN 620 ELSE CALL SOUND(-
100,-5,2):: T=T-1 :: IF T<=0
 THEN 1340
610 CALL W(100)
620 NEXT I
630 DISPLAY AT(20,4):"WELL D
ONE! " :: DISPLAY AT(22,1):"
YOU ARE AT THE POWER STATION
"
640 CALL W(500)
650 DATA aaaaaaaaaaaaaaaa hh
hh_,aaaaaaaaaahhhhh hhhh_,h
hhhhhaaaaaaaa hhhh_,hhhhaa
aaaaaaaa hhhhh_
660 DATA hhhhhhhhaaaaaaa hhh
ha_,hhhhhhhaaaaaaaaa aaa_,h
hhhaaaaaahhhhhhh aa_,hhhaaa
aaaaahhhhh aaa_
670 DATA hhhhhaaaahhh aa
aa_,hhhaaaaaahhh h aaa_,h
hhaaaaaahh hhh aa_,haaaaa
aaaaa hhhhh a_
680 DATA aaaaaaaaa aaaaaa
aa_,aaaaaaaaa hhhhh aaa_,a
aaaaaaa aaaa aaaa_,haaaaa
aa aaaaa haaaa_
690 DATA hhhaaaa aaaaa hhaa
aa_,hhhhaa aaaa hhhhaaa_,h
hhhhaa aaaa hhaaaa_,hhhhha
a ahhhhh aaaaa_
700 DATA hhhhhaa hhhhhh aa
aa_,hhhhhaaa hhhh aaaaa_,h
hhhhaaaa hh aaaaaa_,hhhhaa
aaaa h hhhaaaa_
710 DATA hhhhhaaaa hhhhaa
aa_,hhhhhhaaaa hhhhaaa_,h
hhhhaaaaa h hhhaaa_,hhhhaa
aaaa hhh hhaaa_
720 DATA hhhaaaaaa hhhhhh ha
aa_,hhaaaaaa hhhhhhh aaa_,q
q qqqq qqq qqqqqqqqqq,q
qq qqqq qqqqq qqqqqqqqqq
730 DATA qqqqqqqqqqqqqqqqqq
qqqqqqqq,qqqqqqqqqqqqqqqqqq
qqqqqqqq,,,,,,,,,,,,,
740 REM SHOOT OUT
750 GAME=3 :: CALL Q(GAME,T)
:: CALL SCREEN(6):: CALL CLE
AR :: CALL COLOR(9,2,1,10,2,
1)
760 CALL HCHAR(1,1,99,64):: 
CALL HCHAR(3,1,113,32):: CAL
L HCHAR(6,1,113,32):: CALL H
CHAR(7,1,99,64)
770 FOR V=1 TO 4 :: CALL VCH
AR(1,V,99,8):: NEXT V
780 FOR V=28 TO 32 :: CALL V
CHAR(1,V,99,8):: NEXT V
790 FOR V=7 TO 25 STEP 3 :: 
CALL VCHAR(4,V,113,2):: NEXT
 V
800 N=2 :: FOR R=21 TO 24 :: 
 CALL HCHAR(R,1,103,N):: CAL
L HCHAR(R,33-N,103,N):: N=N+
1 :: NEXT R
810 CALL HCHAR(22,15,103):: 
CALL HCHAR(22,17,103):: CALL
 HCHAR(23,14,103,5):: CALL H
CHAR(24,13,103,7):: CALL HCH
AR(23,16,105)
820 A,B,C,D,HIT=1
830 GOSUB 1050
840 T=T-1 :: IF A=0 THEN 850
 ELSE COL=15 :: GOSUB 970
850 GOSUB 1050 :: FIRE=0 :: 
IF HIT=0 THEN A=0 :: HIT=1
860 IF A=0 AND B=0 AND C=0 A
ND D=0 THEN 1070
870 T=T-1 :: IF B=0 THEN 890
 ELSE COL=5 :: GOSUB 970
880 FIRE=0 :: IF HIT=0 THEN
B=0 :: HIT=1
890 GOSUB 1050 :: IF A=0 AND
 B=0 AND C=0 AND D=0 THEN 10
70
900 T=T-1 :: IF C=0 THEN 920
 ELSE COL=7 :: GOSUB 970
910 FIRE=0 :: IF HIT=0 THEN
C=0 :: HIT=1 ::
920 GOSUB 1050 :: IF A=0 AND
 B=0 AND C=0 AND D=0 THEN 10
70
930 T=T-1 :: IF D=0 THEN 950
 ELSE COL=13 :: GOSUB 970
940 FIRE=0 :: IF HIT=0 THEN
D=0 :: HIT=1
950 GOSUB 1050 :: IF A=0 AND
 B=0 AND C=0 AND D=0 THEN 10
70
960 GOTO 840
970 CALL COLOR(12,COL,2)
980 FOR CO=1 TO 31
990 CALL HCHAR(7,CO,124):: C
ALL HCHAR(7,CO+1,126):: CALL
 HCHAR(8,CO,125):: CALL HCHA
R(8,CO+1,127)
1000 CALL KEY(2,K,S):: IF K<
>18 OR S=0 OR FIRE=1 THEN 10
30
1010 FIRE=1 :: FOR ROW=22 TO
 9 STEP -1 :: CALL HCHAR(ROW
,16,46):: CALL HCHAR(ROW,16,
32):: NEXT ROW
1020 IF CO=15 OR CO=16 THEN
CALL SCREEN(7):: CALL SOUND(
```

```
200,-5,2):: CALL SCREEN(6)::
 HIT=0 :: CALL HCHAR(7,1,99,
64):: GOTO 1050
1030 CALL HCHAR(7,CO,99,2)::
 CALL HCHAR(8,CO,99,2)
1040 NEXT CO
1050 DISPLAY AT(1,1):"TIME L
EFT = ";T :: IF T<=0 THEN 13
40
1060 RETURN
1070 DISPLAY AT(20,1)ERASE A
LL:"WELL DONE YOU HAVE STOPP
ED  THEM ALL"
1080 CALL W(600)
1090 REM MAZE GAME
1100 GAME=4 :: CALL Q(GAME,T
):: CALL SCREEN(14):: CALL C
OLOR(9,1,1)
1110 DISPLAY AT(1,1)ERASE AL
L:"PLEASE WAIT...."
1120 KR=(INT(RND*5)+1)*2 ::
KC=(RND*24)+4
1130 CALL HCHAR(24,1,99,32):
: CALL VCHAR(1,1,99,24):: CA
LL VCHAR(1,32,99,24)
1140 FOR R=3 TO 21 STEP 2 ::
 FOR C=2 TO 31 :: CALL HCHAR
(R,C,102):: NEXT C :: NEXT R
1150 FOR R=3 TO 21 STEP 2 ::
 C=RND*24+4 :: CALL HCHAR(R,
C,32):: NEXT R
1160 CALL HCHAR(KR,KC,96)
1170 CALL COLOR(9,2,1,12,2,1
)
1180 R=22 :: CALL HCHAR(R,C,
123)
1190 DISPLAY AT(1,1):"TIME L
EFT = ";T :: IF T<=0 THEN 13
40 ELSE CALL KEY(3,K,S):: CA
LL JOYST(2,L,M):: IF S=0 AND
 L=0 AND M=0 THEN 1190
1200 CALL HCHAR(R,C,32):: ST
=ST+1 :: IF INT(ST/6)=ST/6 T
HEN T=T-1
1210 IF K=69 OR M=4 THEN 122
0 ELSE IF K=68 OR L=4 THEN 1
230 ELSE IF K=88 OR M=-4 THE
N 1240 ELSE IF K=83 OR L=-4
THEN 1250 ELSE 1190
1220 R=R-1 :: GOTO 1260
1230 C=C+1 :: GOTO 1260
1240 R=R+1 :: GOTO 1260
1250 C=C-1
1260 CALL GCHAR(R,C,G):: CAL
L HCHAR(R,C,123):: IF G=102
THEN T=T-5 :: CALL SOUND(-10
0,-6,0)
1270 IF G=96 THEN 1290
1280 IF G=99 THEN 1310 ELSE
1190
1290 CALL CLEAR :: PRINT "WE
LL DONE YOU DID IT WITH ";T;
"MINUTES LEFT"
1300 CALL SOUND(500,220,0,44
0,0,659,0):: GOTO 1380
1310 CALL CLEAR :: PRINT "YO
U RAN INTO THE OUTSIDE
                          WA
LL AND KNOCKED YOURSELF
 "
1320 PRINT "OUT. YOUR TIME I
S UP BEFORE
          YOU COME ROUND."
1330 GOTO 1380
1340 CALL CLEAR :: PRINT "YO
U RAN OUT OF TIME"
1350 CALL SOUND(900,-7,0)::
CALL SCREEN(7)
1360 CALL SOUND(1500,-5,0)::
 CALL SCREEN(5)
1370 GOTO 1380
1380 CALL W(1000):: DISPLAY
AT(10,4)ERASE ALL:"PLAY AGAI
N ?" :: DISPLAY AT(15,6):"PR
ESS Y"
1390 CALL KEY(2,K,S):: IF S=
0 THEN 1390 ELSE IF K=18 THE
N 1400 ELSE STOP

1400 RESTORE 650 :: CALL COL
OR(9,2,5,10,2,14,11,2,7,12,2
,12):: GOTO 270
1410 SUB TITLE
1420 CALL SCREEN(16)
1430 DISPLAY AT(10,4)ERASE A
LL:"HB AC DE CA    CA FG AC"
 :: DISPLAY AT(11,4):"HB HI
AJ  F     F AJ HI"
1440 DISPLAY AT(15,7):"HB DE
 HB DE CA KC" :: DISPLAY AT(
16,7):"FL MN HB MN  F IB"
1450 CALL SPRITE(#1,124,2,16
,10,10,0,#2,124,5,16,10,0,10
,#3,124,7,160,230,-10,0,#4,1
24,3,160,230,0,-10)
1460 CALL W(4000):: CALL DEL
SPRITE(ALL)
1470 SUBEND
1480 SUB W(X)
1490 FOR D=1 TO X :: NEXT D
:: SUBEND
1500 SUB Q(GAME,T)
1510 CALL CLEAR :: DISPLAY A
T(1,1):"TIME LEFT = ";T
1520 DISPLAY AT(20,1):"PRESS
 1 FOR STORY
 2 FOR INSTRUCTIONS
 3 TO PLAY"
1530 CALL KEY(0,K,S):: IF S=
0 THEN 1530
1540 IF K<49 OR K>51 THEN 15
30
1550 ON ((K-48)+GAME*3)-3 GO
TO 1570,1730,1560,1810,1870,
1560,1930,2000,1560,2050,211
0,1560
1560 SUBEXIT
1570 REM INSTRUCTIONS
1580 PRINT "YOU ARE LYING IN
 BED WHEN
            THE PHONE RINGS.
 "
1590 PRINT "THIS IS WHAT YOU
 HEAR.
          'AGENT 0007 LIST
EN CAREFULLY     "
1600 PRINT "THE NUCLEAR POWE
R STATION IN
          THE NEXT TOWN HA
S BEEN TAKEN  "
1610 PRINT "OVER BY FOUR ROB
OT WORKERS.
          THEY WILL BLOW I
T UP IN ONE     "
1620 PRINT "HOUR UNLESS SOME
ONE CAN STOP
          THEM. THE KEY TO
 THE REACTOR  "
1630 PRINT "IS IN A SAFE IN
THE FLAT
          ABOVE YOURS. YOU
 MUST OPEN" :: CALL W(6000)
1640 PRINT "THE SAFE, DRIVE
TO THE POWER
          STATION, DISABLE
 THE ROBOTS   "
1650 PRINT "AND FIND YOUR WA
Y TO THE
          REACTOR.IF THIS
IS NOT DONE   "
1660 PRINT "WITHIN ONE HOUR
THE WHOLE
          COUNTRY WILL BE
WIPED OUT.    "
1670 PRINT "WE WISH YOU LUCK
 AGENT 0007'
          WITH THAT THE PH
ONE WENT     "
1680 PRINT "DEAD AND I WAS L
EFT WITH ONE
          HOUR TO 'BEAT TH
E ROBOTS'          CCCCCCC
CCCCCCCC"
1690 PRINT "PRESS 2 FOR INST
RUCTIONS
                3 TO PLAY
 "
1700 CALL KEY(0,K,S):: IF S=
0 THEN 1700
1710 IF K<50 OR K>51 THEN 17
00
1720 ON K-49 GOTO 1730,1560
1730 REM INSTRUCTIONS GAME 1
1740 PRINT :: PRINT :: PRINT
 "YOU ARE IN FRONT OF A SAFE
.     "
1750 PRINT "IT HAS 4EDIFFERE
NT COLOURED
          BUTTONS EACH OF
WHICH MUST    "
1760 PRINT "BE PRESSED ONLY
ONCE AND IN
          THE CORRECT ORDE
R TO OPEN      "
1770 PRINT "THE SAFE. EACH T
RY TAKES ONE
          MINUTE OF YOUR T
IME SO YOU    "
1780 PRINT "MUST BE QUICK.US
E KEYS
                r  b  m
y             "
1790 PRINT "AS THE BUTTONS.
 YOUR TIME
          STARTS WHEN YOU
PRESS A KEY"
1800 CALL KEY(0,K,S):: IF S=
0 THEN 1800 ELSE SUBEXIT
1810 REM STORY        GAME 2
1820 CALL CLEAR :: PRINT "YO
U HAVE THE KEY AND MUST
                         DR
IVE TO THE POWER STATION.
"
1830 PRINT "YOU MAY GO EITHE
R THROUGH OR
          ROUND THE FOREST
 BUT WATCH
          THE TIME."
1840 PRINT :: PRINT "PRESS 2
 FOR INSTRUCTIONS
                         3
 TO PLAY"
1850 CALL KEY(0,K,S):: IF S=
0 OR K<50 OR K>51 THEN 1850
1860 ON K-49 GOTO 1870,1560
1870 CALL CLEAR :: PRINT "TH
E JOURNEY THROUGH THE
                         FO
REST TAKES A MINIMUM OF
"

1880 PRINT "20 MINUTES WHILE
 GOING ROUND
          TAKES 40 MINUTES
. EACH TREE   "
1890 PRINT "OR FENCE HIT COS
TS YOU
          ANOTHER MINUTE S
O BE VERY      "
1900 PRINT "CAREFUL.USE THE
S AND D KEYS
          OR THE JOYSTICKS
.          "
1910 PRINT "PRESS ANY KEY TO
 PLAY"
1920 CALL KEY(0,K,S):: IF S=
1 THEN SUBEXIT ELSE 1920
1930 CALL CLEAR :: PRINT "YO
U HAVE NOW ARRIVED AT THE
                         PO
WER STATION AND YOU MUST
 "
1940 PRINT "SHOOT DOWN THE R
OBOTS IN
          ORDER TO GET INS
IDE.          "
1950 PRINT "EACH ROBOT TAKES
 4 MINUTES
          TO WALK ROUND TH
E STATION.      "
1960 PRINT "SO ONE PASSES YO
U EVERY
          MINUTE.
           "
```

```
1970 PRINT "PRESS 2 FOR INST
RUCTIONS
                3 TO PLAY
"
1980 CALL KEY(0,K,S):: IF K<
50 OR K>51 OR S=0 THEN 1980
1990 ON K-49 GOTO 2000,1560
2000 CALL CLEAR :: PRINT "US
E THE FIRE BUTTON OR THE
                          'Y
' KEY TO SHOOT AT THE
"
2010 PRINT "ROBOTS AS THEY P
ASS YOU.
          YOU CAN ONLY SHO
OT AT EACH    "
2020 PRINT "ROBOT ONCE FOR E
ACH TIME
          THAT IT PASSES Y
OU.           "
2030 PRINT "PRESS ANY KEY TO
 GO ON."
2040 CALL KEY(0,K,S):: IF S
THEN 1560 ELSE 2040
2050 CALL SCREEN(8):: CALL C
LEAR :: PRINT "YOU HAVE FINI
SHED OFF THE
             ROBOTS AND NO
W MUST FIND       "
2060 PRINT "YOUR WAY THROUGH
 THE MAZE OF
          PASSAGES TO THE
REACTOR AND   "
2070 PRINT "USE YOUR KEY TO
SWITCH IT
          OFF. I HOPE THAT
 YOU HAVE         "
2080 PRINT "ENOUGH TIME LEFT
.
          PRESS 2 FOR INST
RUCTIONS
                3 TO PLAY"
2090 CALL KEY(0,K,S):: IF K<
50 OR K>51 OR S=0 THEN 2090
2100 ON K-49 GOTO 2110,1560
2110 CALL SCREEN(8):: CALL C
LEAR :: PRINT "USE THE JOYST
ICKS OR THE
             ARROW KEYS TO
 MOVE ALONG THE "
2120 PRINT "PASSAGES TO THE
REACTOR
          SWITCH. (')
              "
2130 PRINT "EACH 6  STEPS TA
KE 1 MINUTE
          AND EACH WALL TO
UCHED COSTS     "
2140 PRINT "YOU ANOTHER 5 MI
NUTES SO BE
          QUICK AND CAREFU
L             "
2150 PRINT "PRESS ANY KEY TO
 GO ON"
2160 CALL KEY(0,K,S):: IF S<
>0 THEN 1560 ELSE 2160
2170 SUBEND
```

```
**************************************
*  Turn of the Card ***** Philip Ord  *
**************************************
```

Anyone who watches 'that program' on TV will pick this one up straight away! A game of judgement/luck like many card games, and rather addictive. No leggy blondes here to put the cards up, though. Shame!

Turn of the Card

```
100 CALL CLEAR
110 CALL SCREEN(15)
120 FOR I=1 TO 8
130 CALL COLOR(I,5,1)
140 NEXT I
150 PRINT "      TURN OF THE
 CARD": : : :"       AUGUST
  1984": : : :"       BY PH
ILIP ORD": : : :
160 PRINT "   INSTRUCTIONS?
(Y OR N)": : : : : :
170 CALL KEY(0,K,S)
180 IF S=0 THEN 170
190 IF K=89 THEN 210
200 IF K=78 THEN 280 ELSE 17
0
210 CALL CLEAR
220 PRINT "  TRY TO GUESS IF
 THE NEXT": :"  CARD IS HIGH
ER OR LOWER": :"  THAN THE O
NE BEFORE.": :
230 PRINT "  YOU START WITH
250 POUNDS": :"  AND RISK AS
 MANY AS YOU": :"  WISH ON T
HE TURN OF": :
240 PRINT "  EACH CARD.": :"
  THE CARDS ARE DISPLAYED":
:"  IN 3 SETS OF 5 AND YOU":
 :
250 PRINT "  CAN CHANGE THE
FIRST CARD": :"  IN EACH SET
 IF YOU WISH.": :
260 FOR DELAY=1 TO 5000
270 NEXT DELAY
280 CALL CLEAR
290 CALL SCREEN(15)
300 RESTORE 350
310 FOR I=96 TO 126
320 READ A$
330 CALL CHAR(I,A$)
340 NEXT I
350 DATA 00,000000E0E0E0E0E0
,E0E0E0E0E0E0E0E0,E0E0E0E0E0
E00000,0000001F1F1F0000,0000
00F8F8F83838
360 DATA 383838F8F8F80000,00
00003F3F3F3838,3838383F3F3F0
000,000000FCFCFC0000
370 DATA 383838F8F8F83838,00
0000383838383,3838383838383
838,00C0C0F8F8F8C0C0,3838383
F3F3F3838,3838383838380000
380 DATA 000000FEFEFEC6C6,C6
C6C6C6C6C6C6C6,C6C6C6FEFEFE0
000,0000001F1F1F0101
390 DATA 000000FCFCFCC0C0,01
0101010101010,C0C0C0C0C0C0C0
0C0,0101013F3F3F0000,C0C0C0C
0C0C00000,1C1C1C1C1C1C1C1C
400 DATA 38F8F8F8F8F8C0C0,39
3F3F3F3F3F3F3F,0000000E1E3E7
CF8,F0E0C0808080C0E0
410 DATA F0F87C3E1E0E0000
420 CALL CHAR(128,"FFFFFFFFF
FFFFFFF")
430 CALL CHAR(136,"A55AA55A5
AA55AA5")
440 FOR SET=9 TO 12
450 CALL COLOR(SET,7,16)
460 NEXT SET
470 CALL COLOR(13,13,1)
480 CALL COLOR(14,5,16)
490 HSC=0
500 CALL CLEAR
510 T=250
520 BONUS=0
530 BON=0
540 X2=1
550 IF X2=4 THEN 3750
560 FOR I=8 TO 18
570 CALL HCHAR(I,7,128,18)
580 NEXT I
590 CALL HCHAR(6,5,42,22)
600 CALL HCHAR(20,5,42,22)
610 CALL VCHAR(7,5,42,14)
620 CALL VCHAR(7,26,42,14)
630 RESTORE 680
640 FOR I=1 TO 10
650 READ A,B,C
660 CALL VCHAR(A,B,136,C)
670 NEXT I
680 DATA 10,9,3,10,10,3,11,1
2,3,11,13,3,12,15,3,12,16,3
690 DATA 13,18,3,13,19,3,14,
21,3,14,22,3
700 M$="SET "&STR$(X2)
710 FOR I=1 TO LEN(M$)
720 CALL HCHAR(10,I+18,ASC(S
EG$(M$,I,1)))
730 NEXT I
740 M$="SCORE TO BEAT="&STR$
(HSC)
750 FOR I=1 TO LEN(M$)
760 CALL HCHAR(19,I+5,ASC(SE
G$(M$,I,1)))
770 NEXT I
780 G2=0
790 F2=0
800 IF F2=0 THEN 900
810 M$="   CHANGE THIS CARD?
(Y OR N)"
820 FOR I=1 TO LEN(M$)
830 CALL HCHAR(4,I,ASC(SEG$(
M$,I,1)))
840 CALL SOUND(1,2000,10)
850 NEXT I
860 CALL SOUND(100,660,5)
870 CALL KEY(0,K,S)
880 IF S=0 THEN 870
890 IF K=89 THEN 900 ELSE 11
00
900 R=10
910 C=9
920 G2=0
930 FOR I=10 TO 13
940 CALL HCHAR(I,9,128,2)
950 NEXT I
960 CALL HCHAR(4,1,32,32)
970 RANDOMIZE
980 X=INT(RND*13)+1
990 ON X GOSUB 1360,1490,163
0,1770,1910,2050,2190,2330,2
470,2610,2750,2890,3030
1000 G2=1
1010 IF C=21 THEN 1020 ELSE
1080
1020 FOR D=1 TO 400
1030 NEXT D
1040 X2=X2+1
1050 F2=0
1060 CALL CLEAR
1070 GOTO 550
1080 F2=F2+1
1090 IF F2=1 THEN 800
1100 IF F2>1 THEN 1120
1110 CALL HCHAR(4,1,32,32)
1120 CALL HCHAR(21,1,32,32)
1130 CALL HCHAR(22,1,32,32)
1140 M$="TOTAL = "&STR$(T)&"
 POUNDS"
1150 FOR I=1 TO LEN(M$)
1160 CALL HCHAR(22,I+5,ASC(S
EG$(M$,I,1)))
1170 NEXT I
1180 INPUT "YOUR STAKE?":E2
1190 IF E2>T THEN 1200 ELSE
1260
1200 CALL CLEAR
1210 CALL SOUND(200,220,0)
1220 PRINT "   SORRY YOU  IN
PUTTED TO": : :"  MUCH, NOW
YOU MUST START": : :"
ALL OVER AGAIN": : : : : : :
 :
1230 FOR D=1 TO 1000
1240 NEXT D
1250 GOTO 500
1260 CALL HCHAR(20,1,32,160)
1270 M$="(H OR L)?"
1280 FOR I=1 TO LEN(M$)
1290 CALL HCHAR(22,I+3,ASC(S
EG$(M$,I,1)))
1300 NEXT I
1310 CALL SOUND(100,550,5)
1320 C=C+3
1330 CALL KEY(0,K,S)
1340 IF S=0 THEN 1330
1350 IF (K=76)+(K=72)THEN 97
0 ELSE 1330
```

```
1360 REM  PRINT ACE
1370 W=1
1380 CALL HCHAR(R,C,103)
1390 CALL HCHAR(R,C+1,101)
1400 CALL HCHAR(R+1,C,110)
1410 CALL HCHAR(R+1,C+1,106)
1420 CALL HCHAR(R+2,C,111)
1430 CALL HCHAR(R+2,C+1,111)
1440 IF G2=0 THEN 1470 ELSE
1450
1450 IF Z=14 THEN 3160
1460 IF Z<14 THEN 3400
1470 Z=14
1480 RETURN
1490 REM PRINT KING
1500 W=13
1510 CALL HCHAR(R,C,107)
1520 CALL HCHAR(R,C+1,124)
1530 CALL HCHAR(R+1,C,123)
1540 CALL HCHAR(R+1,C+1,125)
1550 CALL HCHAR(R+2,C,111)
1560 CALL HCHAR(R+2,C+1,126)
1570 IF G2=0 THEN 1610 ELSE
1580
1580 IF Z=13 THEN 3160
1590 IF Z<13 THEN 3400
1600 IF Z>13 THEN 3720
1610 Z=13
1620 RETURN
1630 REM PRINT QUEEN
1640 W=12
1650 CALL HCHAR(R,C,103)
1660 CALL HCHAR(R,C+1,101)
1670 CALL HCHAR(R+1.C.108)
1680 CALL HCHAR(R+1,C+1,108)
1690 CALL HCHAR(R+2,C,104)
1700 CALL HCHAR(R+2,C+1,122)
1710 IF G2=0 THEN 1750 ELSE
1720
1720 IF Z=12 THEN 3160
1730 IF Z<12 THEN 3400
1740 IF Z>12 THEN 3720
1750 Z=12
1760 RETURN
1770 REM PRINT JACK
1780 W=11
1790 CALL HCHAR(R,C,115)
1800 CALL HCHAR(R,C+1,116)
1810 CALL HCHAR(R+1,C,117)
1820 CALL HCHAR(R+1,C+1,118)
1830 CALL HCHAR(R+2,C,119)
1840 CALL HCHAR(R+2,C+1,120)
1850 IF G2=0 THEN 1890 ELSE
1860
1860 IF Z=11 THEN 3160
1870 IF Z<11 THEN 3400
1880 IF Z>11 THEN 3720
1890 Z=11
1900 RETURN
1910 REM PRINT TEN
1920 W=10
1930 CALL HCHAR(R,C,107)
1940 CALL HCHAR(R,C+1,112)
1950 CALL HCHAR(R+1,C,108)
1960 CALL HCHAR(R+1,C+1,113)
1970 CALL HCHAR(R+2,C,111)
1980 CALL HCHAR(R+2,C+1,114)
1990 IF G2=0 THEN 2030 ELSE
2000
2000 IF Z=10 THEN 3160
2010 IF Z<10 THEN 3400
2020 IF Z>10 THEN 3720
2030 Z=10
2040 RETURN
2050 REM PRINT NINE
2060 W=9
2070 CALL HCHAR(R,C,103)
2080 CALL HCHAR(R,C+1,101)
2090 CALL HCHAR(R+1,C,104)
2100 CALL HCHAR(R+1,C+1,106)
2110 CALL HCHAR(R+2,C,100)
2120 CALL HCHAR(R+2,C+1,102)
2130 IF G2=0 THEN 2170 ELSE
2140
2140 IF Z=9 THEN 3160
2150 IF Z<9 THEN 3400
2160 IF Z>9 THEN 3720
2170 Z=9
2180 RETURN
2190 REM PRINT EIGHT
2200 W=8
2210 CALL HCHAR(R,C,103)
2220 CALL HCHAR(R,C+1,101)
2230 CALL HCHAR(R+1,C,110)
2240 CALL HCHAR(R+1,C+1,106)
2250 CALL HCHAR(R+2,C,104)
2260 CALL HCHAR(R+2,C+1,102)
2270 IF G2=0 THEN 2310 ELSE
2280
2280 IF Z=8 THEN 3160
2290 IF Z<8 THEN 3400
2300 IF Z>8 THEN 3720
2310 Z=8
2320 RETURN
2330 REM PRINT SEVEN
2340 W=7
2350 CALL HCHAR(R,C,100)
2360 CALL HCHAR(R,C+1,101)
2370 CALL HCHAR(R+1,C,96)
2380 CALL HCHAR(R+1,C+1,108)
2390 CALL HCHAR(R+2,C,96)
2400 CALL HCHAR(R+2,C+1,111)
2410 IF G2=0 THEN 2450 ELSE
2420
2420 IF Z=7 THEN 3160
2430 IF Z<7 THEN 3400
2440 IF Z>7 THEN 3720
2450 Z=7
2460 RETURN
2470 REM PRINT SIX
2480 W=6
2490 CALL HCHAR(R,C,103)
2500 CALL HCHAR(R,C+1,105)
2510 CALL HCHAR(R+1,C,110)
2520 CALL HCHAR(R+1,C+1,101)
2530 CALL HCHAR(R+2,C,104)
2540 CALL HCHAR(R+2,C+1,102)
2550 IF G2=0 THEN 2590 ELSE
2560
2560 IF Z=6 THEN 3160
2570 IF Z<6 THEN 3400
2580 IF Z>6 THEN 3720
2590 Z=6
2600 RETURN
2610 REM PRINT FIVE
2620 W=5
2630 CALL HCHAR(R,C,103)
2640 CALL HCHAR(R,C+1,105)
2650 CALL HCHAR(R+1,C,104)
2660 CALL HCHAR(R+1,C+1,101)
2670 CALL HCHAR(R+2,C,100)
2680 CALL HCHAR(R+2,C+1,102)
2690 IF G2=0 THEN 2730 ELSE
2700
2700 IF Z=5 THEN 3160
2710 IF Z<5 THEN 3400
2720 IF Z>5 THEN 3720
2730 Z=5
2740 RETURN
2750 REM PRINT FOUR
2760 W=4
2770 CALL HCHAR(R,C,107)
2780 CALL HCHAR(R,C+1,96)
2790 CALL HCHAR(R+1,C,108)
2800 CALL HCHAR(R+1,C+1,96)
2810 CALL HCHAR(R+2,C,104)
2820 CALL HCHAR(R+2,C+1,109)
2830 IF G2=0 THEN 2870 ELSE
2840
2840 IF Z=4 THEN 3160
2850 IF Z<4 THEN 3400
2860 IF Z>4 THEN 3720
2870 Z=4
2880 RETURN
2890 REM PRINT THREE
2900 W=3
2910 CALL HCHAR(R,C,100)
2920 CALL HCHAR(R,C+1,101)
2930 CALL HCHAR(R+1,C,100)
2940 CALL HCHAR(R+1,C+1,106)
2950 CALL HCHAR(R+2,C,100)
2960 CALL HCHAR(R+2,C+1,102)
2970 IF G2=0 THEN 3010 ELSE
2980
2980 IF Z=3 THEN 3160
2990 IF Z<3 THEN 3400
3000 IF Z>3 THEN 3720
3010 Z=3
3020 RETURN
3030 REM PRINT TWO
3040 W=2
3050 CALL HCHAR(R,C,100)
3060 CALL HCHAR(R,C+1,101)
3070 CALL HCHAR(R+1,C,103)
3080 CALL HCHAR(R+1,C+1,102)
3090 CALL HCHAR(R+2,C,104)
3100 CALL HCHAR(R+2,C+1,105)
3110 IF G2=0 THEN 3140 ELSE
3120
3120 IF Z=2 THEN 3160
3130 IF Z>2 THEN 3720
3140 Z=2
3150 RETURN
3160 CALL SOUND(1000,440,10)
3170 CALL COLOR(13,5,1)
3180 CALL HCHAR(22,1,32,32)
3190 M$="  A PAIR,LOSE HALF
YOUR STAKE"
3200 T=T-E2/2
3210 BON=BON+1
3220 FOR I=1 TO LEN(M$)
3230 CALL HCHAR(22,I,ASC(SEG
$(M$,I,1)))
3240 NEXT I
3250 FOR DELAY=1 TO 400
3260 NEXT DELAY
3270 IF W=1 THEN 1470
3280 IF W=13 THEN 1610
3290 IF W=12 THEN 1750
3300 IF W=11 THEN 1890
3310 IF W=10 THEN 2030
3320 IF W=9 THEN 2170
3330 IF W=8 THEN 2310
3340 IF W=7 THEN 2450
3350 IF W=6 THEN 2590
3360 IF W=5 THEN 2730
3370 IF W=4 THEN 2870
3380 IF W=3 THEN 3010
3390 IF W=2 THEN 3140
3400 IF K=76 THEN 3420
3410 IF K=72 THEN 3560
3420 FOR I=1100 TO 110 STEP
-110
3430 CALL SOUND(100,I,10)
3440 NEXT I
3450 CALL COLOR(13,7,1)
3460 CALL HCHAR(22,1,32,32)
3470 M$="SORRY! THATS WRONG"
3480 FOR I=1 TO LEN(M$)
3490 CALL HCHAR(22,I+4,ASC(S
EG$(M$,I,1)))
3500 NEXT I
3510 FOR DELAY=1 TO 400
3520 NEXT DELAY
3530 T=T-E2
3540 IF T=0 THEN 500
3550 GOTO 3270
3560 FOR I=110 TO 1100 STEP
110
3570 CALL SOUND(100,I,10)
3580 NEXT I
3590 CALL COLOR(13,13,1)
3600 CALL HCHAR(22,1,32,32)
3610 M$="CORRECT!"
3620 T=T+E2
3630 BON=BON+1
3640 IF BON=15 THEN 3650 ELS
E 3660
3650 BONUS=5000
3660 FOR I=1 TO LEN(M$)
3670 CALL HCHAR(22,I+10,ASC(
SEG$(M$,I,1)))
3680 NEXT I
3690 FOR DELAY=1 TO 400
3700 NEXT DELAY
3710 GOTO 3270
3720 IF K=76 THEN 3560
3730 IF K=72 THEN 3420
3740 GOTO 3270
3750 CALL CLEAR
3760 IF T>HSC THEN 3770 ELSE
 3790
3770 HSC=T+BONUS
3780 GOSUB 3880
3790 PRINT "  YOU SCORED ";T
;" ": :"": :"  PLUS A BONUS
OF ";BONUS;"": :"  FOR A TOT
AL OF ";T+BONUS;"": :
```


2021: LINES 3270 TO 3390- CONSIDER INSTEAD:
3270 ON W GOTO 1470,3140,301
0,2870,2730,2590,2450,2310,2
170,2030,1890,1750,1610,3400

```
3800 PRINT "  THE BEST SCORE
 SO FAR": :"  IS ";HSC;" BY
";B$;"": :
3810 PRINT : :
3820 PRINT "  PLAY AGAIN? (Y
 OR N)": :
3830 CALL KEY(0,K,S)
3840 IF S=0 THEN 3830
3850 IF K=89 THEN 500
3860 IF K=78 THEN 3870 ELSE
3830
3870 END
3880 REM HIGHEST SCORE
3890 PRINT "  WELL DONE,YOU'
VE SCORED": :"  THE HIGHEST
SO FAR THIS": :"  SESSION":
:"  WHAT IS YOUR NAME": :
3900 INPUT B$
3910 PRINT : :
3920 RETURN
```

```
****************************************
* Loan Calculations ********** R.King *
****************************************
```

TI BASIC

This program is what they call ´menu-driven´. That has nothing to do with meals on wheels, rather that you are offered a choice of options from a menu. All the options have to do with Loan Calculations:

1) Value of payments
2) Number of payments
3) Balance remaining.

Using the information that you already know, eg. how much you want to borrow and over what period of time, you can then work out how much the individual instalments will be according to an interest factor that you also input.

You will have to ´break´ the program to finish (FCTN/4), but a small program amendment will give you that option at keystroke also.

```
100 CALL CLEAR
110 CALL SCREEN(16)
120 FOR A=1 TO 10
130 CALL COLOR(A,16,7)
140 NEXT A
150 S=0
160 CALL CHAR(130,"3C2020702
070203E")
170 R=13
180 RT=4
190 M$="MENU"
200 A$="VALUE OF PAYMENTS  (
1)"
210 B$="NUMBER OF PAYMENTS (
2)"
220 C$="BALANCE REMAINING  (
3)"
230 D$="CHOOSE OPTIONS 1,2 O
R 3"
240 FOR I=1 TO LEN(M$)
250 CALL HCHAR(5,R,ASC(SEG$(
M$,I,1)))
260 R=R+1
270 NEXT I
280 FOR I=1 TO LEN(B$)
290 CALL HCHAR(7,RT,ASC(SEG$
(A$,I,1)))
300 RT=RT+1
310 NEXT I
320 RT=4
330 FOR I=1 TO LEN(B$)
340 CALL HCHAR(9,RT,ASC(SEG$
(B$,I,1)))
350 RT=RT+1
360 NEXT I
370 RT=4
380 FOR I=1 TO LEN(C$)
390 CALL HCHAR(11,RT,ASC(SEG
$(C$,I,1)))
400 RT=RT+1
410 NEXT I
420 RT=3
430 FOR I=1 TO LEN(D$)
440 CALL HCHAR(22,RT,ASC(SEG
$(D$,I,1)))
450 RT=RT+1
460 NEXT I
470 CALL KEY(0,K,S)
480 IF S=0 THEN 470
490 IF K<>49 THEN 500 ELSE 5
10
500 IF K<>50 THEN 970 ELSE 7
10
510 REM OPTION 1
520 L=0
530 GOSUB 1610
540 PRINT "THIS OPTION WILL
GIVE YOU  THE REPAYMENT VAL
UE"
550 PRINT : : : : :
560 GOSUB 1440
570 GOSUB 1480
580 GOSUB 1510
590 P=LOG(1+(I/1200))*(-N)
600 Q=EXP(P)
610 PMTI=(INT(PV/((1-Q)/(I/1
200))*100))/100
620 IF L=1 THEN 830 ELSE 630
630 IF L=2 THEN 1300 ELSE 64
0
640 IF L=3 THEN 1120 ELSE 65
0
650 PRINT "YOUR PAYMENT IS
":PMTI
660 CALL HCHAR(23,3,130,1)
670 GOSUB 1540
680 CALL KEY(0,K,S)
690 IF S=1 THEN 700 ELSE 680
700 IF K=89 THEN 510 ELSE 10
0
710 REM 2nd OPTION
720 GOSUB 1610
730 S=0
740 W=0
750 PRINT "TO FIND THE NUMBE
R OF       PAYMENTS NECESSAR
Y TO       PAY A LOAN "
760 PRINT : : : : :
770 GOSUB 1440
780 GOSUB 1480
790 GOSUB 1570
800 L=1
810 N=600
820 GOTO 590
830 IF PMTI>PMT THEN 840 ELS
E 870
840 PRINT "THE PAYMENT WILL
NOT PAY OFFTHE LOAN THE MINI
MUM PAYMENTTO PAY OFF THE LO
AN IS      ";PMTI
850 CALL HCHAR(23,3,130,1)
860 GOTO 790
870 N=(INT(10*(LOG(PMT/(PMT-
(PV*I/100/12)))/LOG(1+(I/100
/12)))))/10
880 L=0
890 IF W=1 THEN 900 ELSE 910
900 RETURN
910 PRINT "THE NUMBER OF MON
THS TO PAY OFF THE LOAN IS
":N
920 W=0
930 GOSUB 1540
940 CALL KEY(0,K,S)
950 IF S=1 THEN 960 ELSE 940
960 IF K=89 THEN 710 ELSE 10
0
970 REM 3rd OPTION
980 GOSUB 1610
990 S=0
1000 PRINT "TO DETERMINE THE
 AMOUNT OF  THE LOAN REMAINI
NG "
1010 PRINT : : : : :
1020 INPUT "WHAT WAS THE VAL
UE OF THE   ORIGINAL LOAN
":PV
1030 CALL HCHAR(23,18,130,1)
1040 PRINT : :
1050 GOSUB 1570
1060 PRINT : :
1070 GOSUB 1480
1080 INPUT "HOW MANY PAYMENT
S HAVE BEEN MADE  ":NT
1090 L=3
1100 N=600
1110 GOTO 590
1120 IF PMTI>PMT THEN 1130 E
LSE 1180
1130 PRINT : :
1140 PRINT "THE PAYMENT YOU
ARE MAKING  WILL NOT PAY OFF
 THE LOAN INLESS THAN 50 YEA
RS THE    "
1150 PRINT "MINIMUM PAYMENT
IS ";PMTI
1160 CALL HCHAR(23,22,130,1)
1170 GOTO 1400
1180 W=1
1190 L=0
1200 PRINT : :
1210 GOSUB 870
1220 IF NT>N THEN 1230 ELSE
1270
1230 PRINT : :
1240 PRINT "THE NUMBER OF PA
YMENTS MADE HAS ALREADY PAID
 OFF THE    LOAN YOU PAID OF
F THE LOAN  "
1250 PRINT "AFTER ";N;" PAYM
ENTS"
1260 GOTO 1400
1270 L=2
1280 N=600
1290 GOTO 590
1300 IF PMTI>PMT THEN 1310 E
LSE 1330
1310 PRINT "THE PAYMENT YOU
```

```
ARE MAKING  WILL NOT PAY OFF
 THE LOAN INLESS THAN 50 YEA
RS"
1320 GOTO 1400
1330 P=LOG(1+(I/1200))*(-NT)
1340 L=0
1350 Q=EXP(P)
1360 BAL=(INT(((PV-PMT*((1-Q
))/(I/1200)))/Q)*100)/100)
1370 PRINT : :
1380 PRINT "THE BALANCE OF T
HE LOAN IS  ":BAL
1390 CALL HCHAR(23,3,130,1)
1400 GOSUB 1540
1410 CALL KEY(0,K,S)
1420 IF S=1 THEN 1430 ELSE 1
410
1430 IF K=89 THEN 970 ELSE 1
00
1440 INPUT "WHAT IS THE OUTS
TANDING LOAN ":PV
1450 CALL HCHAR(23,3,130,1)
1460 PRINT : :
1470 RETURN
1480 INPUT "WHAT IS THE YEAR
LY INTEREST RATE AS A PERCEN
TAGE ":I
1490 PRINT : :
1500 RETURN
1510 INPUT "HOW MANY MONTHLY
 PAYMENTS    ":N
1520 PRINT : :
1530 RETURN
1540 PRINT : : : : :
1550 PRINT "DO YOU WISH TO R
ECALCULATE   Y OR N"
1560 RETURN
1570 INPUT "WHAT IS THE PAYM
ENT  ":PMT
1580 CALL HCHAR(23,23,130,1)
1590 PRINT : :
1600 RETURN
1610 CALL CLEAR
1620 FOR B=1 TO 10
1630 CALL SCREEN(6)
1640 CALL COLOR(B,2,6)
1650 NEXT B
1660 CALL COLOR(3,7,16)
1670 CALL COLOR(4,7,16)
1680 CALL COLOR(2,7,16)
1690 CALL COLOR(13,7,16)
1700 RETURN
```

****
** Scrolling Utility - Graham Marshall*
****

Minimemory
or Editor/Assembler/32k/Disk

That clever young man has been at it again - this time a routine to send your monitor crazy! If you have any of the above configurations, you can type this assembly language routine in and have fast scrolling left/right/up/down of the whole screen.

"This program enables you to scroll the screen in any of four directions from Basic using the Minimemory. Although it is already possible to scroll the screen by using the PRINT command, the scroll 'up' command is added for completeness.

The scrolling can be accessed by the following commands which can be entered either in a program or by themselves.

CALL LINK("SUP") - scrolls the screen up 1 line and fills the bottom line with spaces.

CALL LINK("SDOWN") - scrolls the screen down 1 line and fills the top line with spaces.

CALL LINK("SLEFT") - scrolls the screen to the left 1 column filling the rightmost column with spaces.

CALL LINK("SRIGHT") - scrolls the screen to the right 1 column filling the leftmost column with spaces.

To use the program, load the assembler into Minimemory using the Easy Bug 'L' command, run the program NEW, and type in this listing exactly as it is found.

eg. to type the first line press:

<SPACE> AORG <SPACE> 7D00 <ENTER>

etc.

After typing is completed press QUIT. Now enter Easy Bug and press:

S 7000 <ENTER> (to save to tape)

The prompt 'TO?' will appear on the screen. Then press>

7FFF <ENTER>

Now the usual procedure for saving programs will happen. To reload use Easy Bug's command.

To use the program you must enter Basic. If the program was entered

correctly you will now be able to use the CALL LINK commands described above.

MINIMEMORY VERSION:

```
AORG >7D00
LI   R0,704
LI   R1,BF
LI   R2,32
BLWP @>6030
AI   R0,32
LI   R1,BF
BLWP @>6028
AI   R0,-64
CI   R0,-32
JNE  >7D04
LI   R0,0
LI   R1,>2000
BLWP @>6024
INC  R0
CI   R0,33
JNE  >7D2A
B    *R11
LI   R0,>0020
LI   R1,BF
LI   R2,32
BLWP @>6030
AI   R0,-32
LI   R1,BF
BLWP @>6028
AI   R0,64
CI   R0,768
JNE  >7D40
LI   R0,736
LI   R1,>2000
BLWP @>6024
INC  R0
CI   R0,768
JNE  >7D66
B    *R11
CLR  R0
LI   R1,BF
LI   R2,31
BLWP @>6030
AI   R0,1
LI   R1,BF
BLWP @>6028
```

```
     AI   R0,31
     CI   R0,768
     JNE  >7D7A
     LI   R0,0
     LI   R1,>2000
     BLWP @>6024
     AI   R0,32
     CI   R0,768
     JNE  >7DA0
     B    *R11
     LI   R0,1
     LI   R1,BF
     LI   R2,31
     BLWP @>6030
     AI   R0,-1
     LI   R1,BF
     BLWP @>6028
     AI   R0,33
     CI   R0,769
     JNE  >7DBB
     LI   R0,31
     LI   R1,>2000
     BLWP @>6024
     AI   R0,32
     CI   R0,799
     JNE  >7DDE
     B    *R11
BF   BSS  32
     AORG >701C
     DATA >7E12
     DATA >7FC8
     DATA >7FC8
     TEXT 'SDOWN '
     DATA >7D00
     TEXT 'SUP   '
     DATA >7D3C
     TEXT 'SRIGHT'
     DATA >7D78
     TEXT 'SLEFT '
     DATA >7DB4
     END
```

ED/ASS LOAD FROM BASIC:

```
- CALL LOAD("DSK1.SCROLL-E/A") -
```

```
     DEF  SDOWN,SUP,SRIGHT,SLEFT
     REF  VMBW,VMBR,VSBW
SDOWN LI  R0,704
LOOP1 LI  R1,BUF
     LI   R2,32
     BLWP @VMBR
     AI   R0,32
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,-64
     CI   R0,32
     JNE  LOOP1
     LI   R0,0
     LI   R1,>2000
NEXT1 BLWP @VSBW
     INC  R0
     CI   R0,33
     JNE  NEXT1
     RT
SUP  LI   R0,>0020
LOOP2 LI  R1,BUF
     LI   R2,32
     BLWP @VMBR
     AI   R0,-32
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,64
     CI   R0,768
     JNE  LOOP2
     LI   R0,736
     LI   R1,>2000
NEXT2 BLWP @VSBW
     INC  R0
     CI   R0,768
     JNE  NEXT2
     RT
SRIGHT LI R0,>0000
LOOP3 LI  R1,BUF
     LI   R2,31
     BLWP @VMBR
     AI   R0,1
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,31
     CI   R0,768
     JNE  LOOP3
     LI   R0,0
     LI   R1,>2000
NEXT3 BLWP @VSBW
     AI   R0,32
     CI   R0,768
     JNE  NEXT3
     RT
SLEFT LI  R0,>0001
LOOP4 LI  R1,BUF
     LI   R2,31
     BLWP @VMBR
     AI   R0,-1
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,33
     CI   R0,769
     JNE  LOOP4
     LI   R0,31
     LI   R1,>2000
NEXT4 BLWP @VSBW
     AI   R0,32
     CI   R0,799
     JNE  NEXT4
     RT
BUF  BSS  32
     END
```

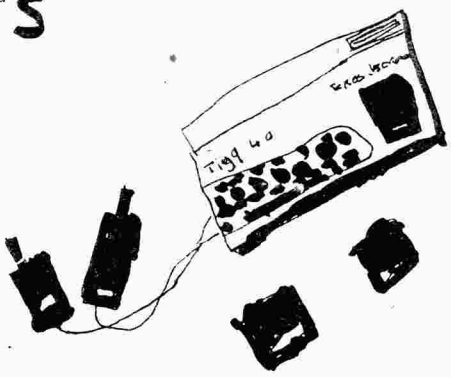
ED/ASS LOAD FROM EB:

```
- CALL LOAD("DSK1.SCROLL-EB") -
```

```
     DEF  SDOWN,SUP,SRIGHT,SLEFT
VSBW EQU  >2020
VMBW EQU  >2024
VMBR EQU  >202C
SDOWN LI  R0,704
LOOP1 LI  R1,BUF
     LI   R2,32
     BLWP @VMBR
     AI   R0,32
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,-64
     CI   R0,32
     JNE  LOOP1
     LI   R0,0
     LI   R1,>2000
NEXT1 BLWP @VSBW
     INC  R0
     CI   R0,33
     JNE  NEXT1
     RT
SUP  LI   R0,>0020
LOOP2 LI  R1,BUF
     LI   R2,32
     BLWP @VMBR
     AI   R0,-32
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,64
     CI   R0,768
     JNE  LOOP2
     LI   R0,736
     LI   R1,>2000
NEXT2 BLWP @VSBW
     INC  R0
     CI   R0,768
     JNE  NEXT2
     RT
SRIGHT LI R0,>0000
LOOP3 LI  R1,BUF
     LI   R2,31
     BLWP @VMBR
     AI   R0,1
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,31
     CI   R0,768
     JNE  LOOP3
     LI   R0,0
     LI   R1,>2000
NEXT3 BLWP @VSBW
     AI   R0,32
     CI   R0,768
     JNE  NEXT3
     RT
SLEFT LI  R0,>0001
LOOP4 LI  R1,BUF
     LI   R2,31
     BLWP @VMBR
     AI   R0,-1
     LI   R1,BUF
     BLWP @VMBW
     AI   R0,33
     CI   R0,769
     JNE  LOOP4
     LI   R0,31
     LI   R1,>2000
NEXT4 BLWP @VSBW
     AI   R0,32
     CI   R0,799
     JNE  NEXT4
     RT
BUF  BSS  32
     END
```

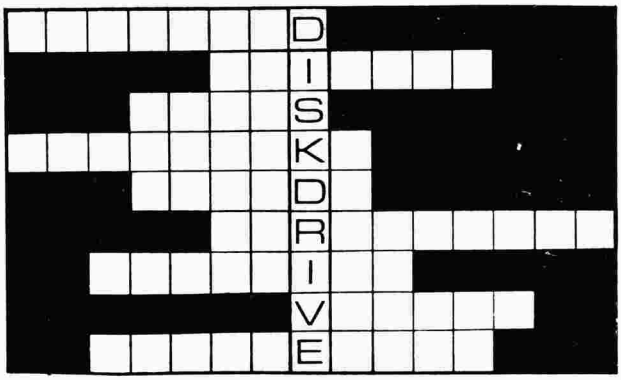
"LITTLE BITS"

99/4a Magazine
Parco Electrics
2 Devonshire Court
Heathpark
Honiton
Devon



At long last we are getting a little bit more interest in LITTLE BITS!

Here follows a puzzle sent in by Kathryn Finch (Age 11) of Sampford Peverell in Devon. (proves we do have local customers as well!) There are no prizes, it's strictly for fun, and the answers will be found elsewhere in this mag.



You type on it

Used instead of writing

Put in cassette recorders

Used for games

You listen to them

Plug in games

You can make up pictures with them

Everything .....es when you type call clear

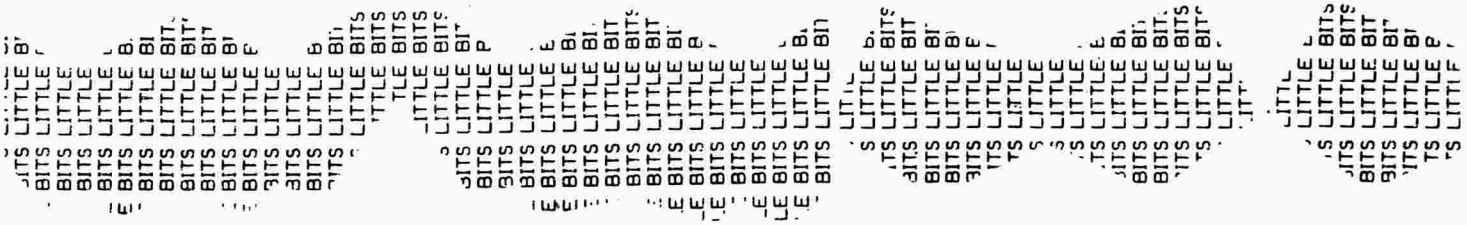
Another memory

And here's a novel program that Kathryn wrote, called 'Watchit'. You will be shown several objects for a few moments, and then when the screen clears you will be asked questions about what you have seen. Yes, it IS harder than it sounds, which is why you must take care to 'Watchit'!

```
100 REM WATCHIT
110 CALL CLEAR
120 PRINT "***********"
130 PRINT "*         *"
140 PRINT "* WATCHIT *"
150 PRINT "*         *"
160 PRINT "***********"
170 PRINT
180 PRINT
190 PRINT "DO YOU WANT INSTR
UCTIONS?"
200 PRINT "         Y OR N
    "
210 INPUT A$
220 IF A$="Y" THEN 230 ELSE
400
230 CALL CLEAR
240 PRINT "INSTRUCTIONS"
250 PRINT
260 PRINT "WATCHIT IS A GAME
 WHERE YOU"
270 PRINT "CAN SEE IF YOU AR
E GOOD AT"
280 PRINT "NOTICING THINGS."
: :"A SCREEN WITH TWELVE LIT
TLE"
290 PRINT "PICTURES DISPLAYE
D WILL"
300 PRINT "APPEAR.STUDY IT C
AREFULLY!": :"NOW YOU WILL B
E ASKED"
310 PRINT "QUESTIONS ON SOME
 OF THE"
320 PRINT "PICTURES YOU SEE.
": :
330 PRINT "DEPENDING ON THE
NUMBER YOU"
340 PRINT "GET RIGHT YOU WIL
L SCORE"
350 PRINT "MARKS OUT OF TEN.
"
360 FOR DELAY=1 TO 3000
370 NEXT DELAY
380 PRINT
390 PRINT
400 CALL CLEAR
410 CALL SCREEN(16)
420 B$="FFFFFFFFFFFFFFFF"
430 CALL CHAR(128,B$)
440 CALL COLOR(13,6,6)
450 CALL HCHAR(2,3,128)
460 CALL HCHAR(4,6,128)
470 CALL HCHAR(5,5,128)
480 CALL HCHAR(6,4,128)
490 CALL HCHAR(7,3,128)
500 CALL HCHAR(3,7,128)
510 CALL HCHAR(2,7,128)
520 CALL HCHAR(1,3,128,5)
530 CALL HCHAR(8,3,128,5)
540 C$="FFFFFFFFFFFFFFFF"
550 CALL CHAR(136,C$)
560 CALL COLOR(14,4,4)
570 CALL HCHAR(2,10,136,6)
580 CALL HCHAR(3,10,136,6)
590 CALL HCHAR(4,10,136,6)
600 CALL HCHAR(5,10,136,6)
610 CALL HCHAR(6,10,136,6)
620 CALL HCHAR(7,10,136,6)
630 CALL HCHAR(9,27,136,3)
640 CALL HCHAR(10,26,136,5)
650 CALL HCHAR(11,25,136,7)
660 CALL HCHAR(12,26,136,5)
670 CALL HCHAR(13,27,136,3)
680 CALL HCHAR(16,12,136,3)
690 CALL HCHAR(14,16,136)
700 CALL HCHAR(15,11,136)
710 CALL HCHAR(15,15,136)
720 CALL HCHAR(14,10,136)
730 CALL VCHAR(14,13,136,2)
740 D$="FFFFFFFFFFFFFFFF"
750 CALL CHAR(144,D$)
760 CALL COLOR(15,11,11)
770 CALL HCHAR(11,13,144)
780 CALL HCHAR(4,17,144,3)
790 CALL HCHAR(5,17,144,3)
800 CALL HCHAR(4,22,144,3)
810 CALL HCHAR(5,22,144,3)
820 CALL VCHAR(1,20,144,8)
830 CALL VCHAR(1,21,144,8)
840 E$="FFFFFFFFFFFFFFFF"
850 CALL CHAR(152,E$)
860 CALL COLOR(16,2,2)
870 CALL HCHAR(1,25,152,8)
880 CALL HCHAR(6,25,152,8)
890 CALL HCHAR(8,25,152,3)
900 CALL HCHAR(8,30,152,3)
910 CALL HCHAR(7,26,152)
920 CALL HCHAR(7,31,152)
930 CALL VCHAR(2,25,152,4)
940 CALL VCHAR(2,32,152,4)
950 F$="00003C3C3C3C0000"
960 CALL CHAR(120,F$)
970 CALL COLOR(12,2,16)
980 CALL VCHAR(2,31,120,4)
990 G$="FFFFFFFFFFFFFFFF"
1000 CALL CHAR(112,G$)
1010 CALL COLOR(11,15,15)
1020 CALL HCHAR(9,4,112,2)
1030 CALL HCHAR(10,3,112,4)
1040 CALL HCHAR(11,2,112,6)
1050 CALL HCHAR(12,1,112,8)
1060 CALL VCHAR(13,4,112,4)
1070 CALL VCHAR(13,5,112,4)
1080 H$="FFFFFFFFFFFFFFFF"
1090 CALL CHAR(104,H$)
1100 CALL COLOR(10,14,14)
1110 CALL HCHAR(9,13,104)
1120 CALL HCHAR(10,12,104)
1130 CALL HCHAR(10,14,104)
1140 CALL HCHAR(11,11,104)
1150 CALL HCHAR(11,15,104)
1160 CALL HCHAR(12,12,104)
1170 CALL HCHAR(12,14,104)
1180 CALL HCHAR(13,13,104)
1190 CALL HCHAR(10,19,104,5)
1200 CALL HCHAR(13,19,104,5)
1210 CALL VCHAR(10,18,104,6)
1220 CALL VCHAR(11,23,104,2)
1230 I$="FFFFFFFFFFFFFFFF"
1240 CALL CHAR(96,I$)
1250 CALL COLOR(9,7,7)
1260 CALL VCHAR(14,28,96,3)
1270 J$="FFFFFFFFFFFFFFFF"
1280 CALL CHAR(56,J$)
```

```
1290 CALL COLOR(4,5,5)
1300 CALL HCHAR(17,4,56,2)
1310 CALL HCHAR(18,3,56,4)
1320 CALL HCHAR(19,2,56,6)
1330 CALL HCHAR(20,1,56,8)
1340 CALL HCHAR(21,1,56,8)
1350 CALL HCHAR(22,2,56,6)
1360 CALL HCHAR(23,3,56,4)
1370 CALL HCHAR(24,4,56,2)
1380 K$="FFFFFFFFFFFFFFFF"
1390 CALL CHAR(37,K$)
1400 CALL COLOR(1,9,16)
1410 CALL VCHAR(18,10,37,6)
1420 CALL VCHAR(18,11,37,6)
1430 CALL VCHAR(18,14,37,6)
1440 CALL VCHAR(18,15,37,6)
1450 CALL VCHAR(17,25,37,7)
1460 CALL VCHAR(17,26,37,7)
1470 CALL VCHAR(20,29,37,5)
1480 CALL VCHAR(20,30,37,5)
1490 CALL VCHAR(18,24,37)
1500 CALL VCHAR(24,19,37)
1510 CALL VCHAR(20,12,37,2)
1520 CALL VCHAR(20,13,37,2)
1530 CALL VCHAR(22,27,37,2)
1540 CALL VCHAR(22,28,37,2)
1550 CALL VCHAR(22,31,37,2)
1560 CALL VCHAR(22,32,37,2)
1570 L$="FFFFFFFFFFFFFFFF"
1580 CALL CHAR(42,L$)
1590 CALL COLOR(2,12,12)
1600 CALL HCHAR(17,20,42,2)
1610 CALL HCHAR(18,20,42,2)
1620 CALL HCHAR(18,17,42)
1630 CALL HCHAR(18,24,42)
1640 CALL HCHAR(19,17,42,8)
1650 CALL HCHAR(20,19,42,4)
1660 CALL HCHAR(21,19,42,4)
1670 CALL HCHAR(22,19,42,4)
1680 CALL VCHAR(23,19,42,2)
1690 CALL HCHAR(23,23,42)
1700 CALL HCHAR(24,24,42)
1710 FOR DELAY=1 TO 5000
1720 NEXT DELAY
1730 CALL CLEAR
1740 PRINT "     QUESTIONS
  ": : :
1750 CALL COLOR(2,2,16)
1760 CALL COLOR(3,2,16)
1770 CALL COLOR(4,2,16)
1780 PRINT
1790 PRINT "HOW MANY SHAPES
WERE THERE?": :
1800 INPUT Q
1810 IF Q=3 THEN 1820 ELSE 1
850
1820 PRINT "CORRECT"
1830 COUNT=1
1840 GOTO 1870
1850 PRINT "WRONG"
1860 GOTO 1870
1870 PRINT "HOW MANY NUMBERS
?": :
1880 INPUT R
1890 IF R=2 THEN 1910 ELSE 1
940
1900 COUNT=COUNT+1
1910 PRINT "CORRECT"
1920 COUNT=COUNT+1
1930 GOTO 1960
1940 PRINT "WRONG"
1950 GOTO 1960
1960 PRINT "HOW MANY BUTTONS
 ON THE TV?": :
1970 INPUT S
1980 IF S=4 THEN 1990 ELSE 2
020
1990 PRINT "CORRECT"
2000 COUNT=COUNT+1
2010 GOTO 2040
2020 PRINT "WRONG"
2030 GOTO 2040
2040 PRINT "WHAT WERE THE LE
TTERS?": :
2050 INPUT T$
2060 IF T$="H AND P" THEN 20
70 ELSE 2100
2070 PRINT "CORRECT"
2080 COUNT=COUNT+1
2090 GOTO 2120
2100 PRINT "WRONG"
2110 GOTO 2120
2120 PRINT "WHAT COLOUR WAS
THE ARROW?": :
2130 INPUT U$
2140 IF U$="GREY" THEN 2150
ELSE 2180
2150 PRINT "CORRECT"
2160 COUNT=COUNT+1
2170 GOTO 2200
2180 PRINT "WRONG"
2190 GOTO 2200
2200 PRINT "WHAT WAS THE MAN
 DOING?": :
2210 INPUT AB$
2220 IF AB$="DANCING" THEN 2
230 ELSE 2260
2230 PRINT "CORRECT"
2240 COUNT=COUNT+1
2250 GOTO 2280
2260 PRINT "WRONG"
2270 GOTO 2280
2280 PRINT "WHAT COLOUR WAS
THE FLOWER?": :
2290 INPUT CD$
2300 IF CD$="PURPLE" THEN 23
10 ELSE 2340
2310 PRINT "CORRECT"
2320 COUNT=COUNT+1
2330 GOTO 2370
2340 PRINT "WRONG"
2350 GOTO 2370
2360 CALL COLOR(4,2,16)
2370 PRINT "HOW MANY MEN WER
E THERE?": :
2380 INPUT Z
2390 IF Z=1 THEN 2400 ELSE 2
430
2400 PRINT "CORRECT"
2410 COUNT=COUNT+1
2420 GOTO 2450
2430 PRINT "WRONG"
2440 GOTO 2450
2450 PRINT "WERE THERE ANY C
IRCLES?": :
2460 INPUT GH$
2470 IF GH$="NO" THEN 2480 E
LSE 2510
2480 PRINT "CORRECT"
2490 COUNT=COUNT+1
2500 GOTO 2530
2510 PRINT "WRONG"
2520 GOTO 2530
2530 PRINT "WHAT COLOUR WAS
THE CROSS?": :
2540 INPUT IJ$
2550 IF IJ$="YELLOW" THEN 25
60 ELSE 2590
2560 PRINT "CORRECT": : :
2570 COUNT=COUNT+1
2580 GOTO 2620
2590 PRINT "WRONG"
2600 GOTO 2620
2610 CALL COLOR(3,2,16)
2620 IF COUNT=10 THEN 2630 E
LSE 2640
2630 PRINT "EXCELLENT-YOU SC
ORED";COUNT;"OUT OF TEN"
2640 IF COUNT<10 THEN 2650 E
LSE 2660
2650 PRINT "BAD LUCK,YOU SCO
RED";COUNT;"OUT OF TEN."
2660 PRINT : :"WOULD YOU LIK
E TO PLAY      AGAIN? YES OR
 NO?"
2670 INPUT ANS$
2680 IF ANS$="YES" THEN 100
ELSE 2690
2690 END
```

"LITTLE BITS"



Jo Ann's Juicy Bits.

Jo Ann's Juicy Bits.

How nice to have an article submitted by a member of the fairer sex. What's more, we have here what might become a series. Jo Ann Copeland, better half of Scott Copeland (whose name you will also find in the pages of 99/4a in this and future editions), gets a lot of help and enjoyment out of her TI. The pair of them are hardened Adventure addicts, and have been known to go to extraordinary lengths to feed their addiction. But Jo Ann has also for some time been having an affair. What Scott is now struggling to come to terms with is the fact that Jo Ann has fallen helplessly in love with her TI-Writer. More and more time is spent with this new friend, and Jo Ann herself confesses that she can't get enough of it.

We at 99/4a, as you know, tend to be a frank and open bunch, and we feel that by giving exposure to Jo Ann's obsession, balance and harmony may be restored to the Copeland household. Perhaps when she has had the chance to get this thing out of her system, Jo Ann will once again realize that the chauffeur chappy is actually her husband Scott; and that in the light of her new-found liberty, Scott himself will be able to discard once and for all the ironing apron and rubber-gloves and go back to work.

We keep everything crossed.

TI-WRITER/WORD PROCESSOR

Part 1

by Jo Ann Copeland.

Greatest thing since the invention of the wheel? Almost. For me, the TI Writer/Word Processor is. Anyone who hears from me via letter knows they will be getting a computer printout off the word processor. No more writing pages over because of spelling errors or because you forgot to say something three paragraphs ago. The TI Writer gives you the advantage of doing these things without wasting paper or getting writer's cramp.

As I assume it is with most people of average income, I started off with a computer system of minimal equipment. Whenever I heard of a special sale on somewhere I'd go shopping and finding computer equipment had to check it out. Naturally this led to buying 'pieces' bit by bit and at this point, accumulating a total system, en masse. The problem then, what to do with it all? I immediately checked out the word processing areas while my husband graduated from TI Basic to TI Machine and Assembly Language, still graduating to Pascal, etc. And myself, well I have to admit I'm still on the word processor, as it is my main interest besides of course playing Adventure games, etc. Still, after using it for several months, I just recently opened up the manual to discover another world of commands and usage.

First, you need to have the TI-Writer word processor cartridge and program diskette. Of course, the TI99/4a console (what would we all do without it?), any monitor or television set (connected to the computer of course), the TI Disk Memory Drive, Disk Controller Card, Memory Expansion Card, and last but not least the RS232 Interface Card, naturally to work with a compatible printer. Having had all this, I set myself up, turned on the disk and computer and got ready to go. I got past the title screen just fine, and found my way to the word processor and hit '1' for Text Editor. Getting a bunch of lines I figured I'd hit a key

and hunt and peck my way through hopefully without messing myself up. 'Read the manual' you say? You've got to be kidding! What could there be to a word processor? Little did I know then what I know now. However, I experimented with one option at a time and slowly but surely found out what each was for. Using the Text Editor, you can create almost anything you wish to type, from letters to friends or businesses, term papers, write books, whatever. After typing, you can edit misteaks, switch whole paragraphs, insert lines, format block typing, and of course save it all for prosterity.

(I think she means posterity - Ed.
I only wish it was prosperity!)

To type, say a letter, choose (T) for tabs. Here you can set up your left margin by pressing (L) at your starting point, pressing the space bar over to where you want your tabs and pressing (T) for tab, and then spacing over again to where you want your right margin to be (R). Various commands will be discussed later for formatting block style and indenting, etc. When finished, hit <ENTER> and you'll see line numbers appear. First line for typing, 0001. Pressing FCTN/7 moves the cursor to your tab marking for typing. Type in a few lines and you'll notice the cursor brings you to the following lines (word wraps) to 0002 then 0003, etc. No need to hit <ENTER> until you want to start a new paragraph. At this point, you should notice that you have seen Line numbers, Cursor, and End-of-File marker. You can ignore the End-of-Line marker, as it will move down as your typed lines do. When typing a longer page than you originally thought your letter would be, you can mark a new page by hitting CTRL/9. This command has a new page start where you wish it to. However, if you leave the command out, the word processor will insert one for you as it only manages a line up to 80 columns (characters) lengthwise and 66 on a page. It will space 4 lines at the top and bottom. If you wish to start a new paragraph on a new page rather than having the computer start within a paragraph you will want to mark a command before the paragraph (CTRL/9). Unfortunately, if your printer does not recognize form-feed characters it will not recognize new page. Fortunately, mine does, and I use it rather than letting the computer do so. When you have finished your document, press FCTN/9 and it will return you to the main Menu screen. Pressing SF (Save File) and DSK2.LETTER (or whatever you want to save your document under) will save the document on the disk for future use.

Jo Ann's Juicy Bits.

BRIGHT SPARKS

We forgot to point out in the last issue that the helpful Bright Sparks article was written by Graham Marshall from somewhere north of England called I think Scotland. Dispels the myth that they all spend the entire time dancing around in skirts with furry purses singing Och Weel an' a Hey Nonnie Noo consuming sheep's innards, porridge with salt rather than sugar, and humping lumps of tree-trunk around, doesn't it?

Having now lost several highland friends I would like to make peace by introducing another excellent effort by Graham Marshall.

The STATUS REGISTER

One thing which seems to confuse many budding Assembly Language programmers is the Status Register. This article will cover the use of the status register during comparisons and conditional jump instructions, hopefully explaining how it is changed and used during the execution of these instructions.

Firstly, here is a diagram and table showing what each bit of the status register does. Don't worry if you are baffled by this just now, it should become clearer later in this article.

0 L>	1 A>	2 EQ	3 C	4 OV	5 OP	6 X	7	8	9	10	11	12 IN	13 T	14 MA	15 SK
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Imagine the above diagram being the status register. This is what it would look like when the computer is switched on with each of its bits set to 0. This next table shows the names of the 16 bits and their use.

BIT NO.	NAME	USE
0	L >	SET TO 1 IF LOGICAL GREATER THAN
1	A >	SET TO 1 IF ARITHMETICAL GREATER THAN
2	EQ	SET TO 1 IF EQUAL
3	C	SET TO 1 IF CARRY OF BIT 0
4	OV	SET TO 1 IF AN OVERFLOW OCCURS
5	OP	SET TO 1 IF ODD PARITY
6	X	SET TO 1 IF EXTENDED OPERATION
7-11		THESE BITS ARE RESERVED
12-15	INT MASK	INTERUPT MASK

This article is only concerned with the L>, A> and EQU bits of the status register, so the rest may be ignored for now. From now on in this article if one of these three bits is changed to 1 I will refer to it has being set, and if one of these bits is changed to 0 I will refer to it as being reset.

All this is leading up to the fact that when you make a comparison, the status register is changed, depending on the result of the comparison.

The C command compares the contents of two workspace registers, so if:

R0 contained >12 and R1 >3F then A>, L> and EQU would be reset.

R0 contained >3F and R1 >3F then EQ would be set, A> and L> reset.

R0 contained >42 and R1 >3F then A>, L> would be set, EQ being reset.

These are the compare commands which are commonly used:

CI compares a workspace register with a number. eg CI R0,3 would compare the contents of R0 with 3 and adjust the status register accordingly.

CB compares the leftmost (most significant byte) of two workspace registers. eg CB R1,R2 if R1 contained >3F7B and R2 >3B8C then >3F would be compared to >3B, adjusting the status register.

You may have wondered why there are two bits for 'greater than' type comparisons, called L> and A>. To the computer, >FFFF is logically greater than >0001, but not arithmetically. Why, you may ask, well it is due to the way the computer stores negative numbers.

>0001 in binary is:

0000 0000 0000 0001

To make it negative, you must take its complement (or inverse):

1111 1111 1111 1110

and add 1, giving:

1111 1111 1111 1111

If R0 contained >FFFF then CI R0, >0001 would set L> and reset A>.

If R0 contained >0001 then CI R0, >FFFF would set A> and reset L>.

If you added >0001 and >FFFF you would get >0000 (the C bit of the status register would also be set)

This effectively gives you the choice of two number systems. The two's complement system of integers between -32768 and +32768, and integers between 0 and 65536.

Conditional jump instructions jump depending on the contents of the status register, eg:

JHE LP will jump to the line beginning with label LP if EQ or L> are set.

Here is a table showing the jump instructions which are dependent on the L>, A>, and EQU bits of the status register.

Command Name	Instruction	L>	A>	EQ	Jump if
Jump if High	JH	✓		✓	L>=1 and EQ=0
Jump if Low	JL	✓		✓	L>=0 and EQ=0
Jump if High or Equal	JHE	✓		✓	L>=1 or EQ=1
Jump if Low or Equal	JLE	✓		✓	L>=0 or EQ=1
Jump if Greater Than	JGT		✓		A>=1
Jump if Less Than	JLT		✓	✓	A>=0 and EQ=0
Jump if Not Equal	JNE			✓	EQ=0
Jump if Equal	JEQ			✓	EQ=1

If you have not fully grasped what I have said yet, this short program and its line by line explanation should clear things up.

```
    CLR R0
LP  INC R0
    CI R0,30000
    JNE LP
    B *R11
```

CLR R0: The CLR command stands for CLear Register. In practise it sets every bit of R0 to 0. This is the same as LI R0,0 only it uses two less bytes thus saving valuable memory space.

LP INC R0: Firstly, this line begins with a label, so that the line may be accessed again easily later in the program. INC stands for INCrease, and will add 1 to register 0 (R0) in this program. Generally it can be used to add 1 to the contents of any workspace register.

CI R0,30000: CI stands for Compare Immediate, and will compare the contents of R0 to 30000 in this program, changing the status register depending on the result of the comparison. If R0 contains a number less than 30000 then EQ, L> and A> would be reset. If R0 contained 30000 then then EQ would be set, L>, A> being reset. If R0 contained a number greater than 30000 then A>, L> would be set, EQ being reset.

(Note: In this program when R0 reaches 30000 it stops, so A> and L> will never be set during it.)

JNE LP: JNE stands for Jump if Not Equal. This means that it will jump to the line beginning with the label LP if EQ=0, if EQ=1 the next statement will be run. (EQ will equal 1 when R0 contains 30000). If JNE is replaced with JL, the program will halt when R0 reaches 30001 (when L> is set).

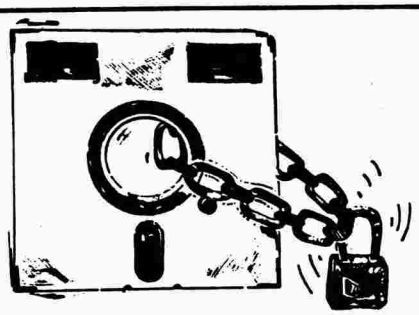
B *R11: will return to EASY BUG (or the calling program).

As a matter of interest, and to give you an idea of how fast assembly language is, this basic program will do near enough the same thing, taking over a minute:

```
10 FOR A=1 TO 30000
20 NEXT A
```

I hope this article has added to your knowledge of 9900 Assembly Language.

Graham Marshall.



TI LIST/RUN PROTECTION.
in Extended Basic.

SCOTT COPELAND is an amiable American TI enthusiast (see JO ANN'S JUICY BITS!) who is serving in the U.K. at RAF Lakenheath. He wrote with some advice about program protecting, as he found that many people didn't even realize it was possible.

List Protector:

The TI List and Run Protector is not only very interesting but also very useful. It will prevent anyone from listing or attempting to change program contents. It will also prevent anyone from saving it to another disk or cassette using another name. It seems to be failsafe as far as protecting your program contents, program ideas, and data files. However, it is not failsafe when it comes to pirating. Cassettes can be reproduced by tape to tape method and diskettes can be copied with the Disk Manager. Even though they can pirate your program, they still cannot list it or write to your program. Now that you know about it, here is the method:

Save your programs like this:

cassettes: SAVE CS1,PROTECTED

disks: SAVE DSKn.NAME,PROTECTED
(n - drive number
NAME - program name)

When someone tries to list or write or attempts to save the program using a different name or, for that matter, your program name, they will get this response displayed on their screen:

PROTECTION VIOLATION

Run protector:

To keep people from running your prgrams you can insert the following at the beginning of your programs:

```
100 INPUT "ENTER ACCESS CODE":A$
110 IF A$="CODE" THEN 130 ELSE 12
120 END
130 (beginning of program)
```

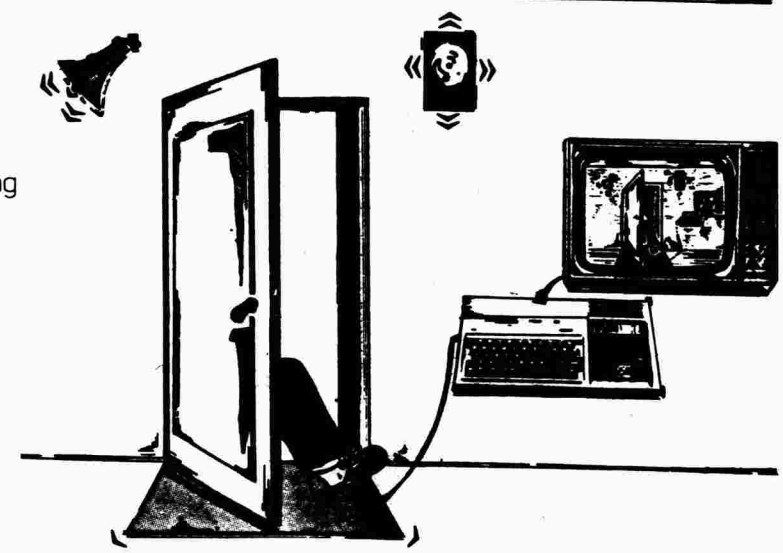
CODE = any desired combination of letters or numbers - you can also conceal them by holding the Control Key while depressing the letters. This will allow only you to realize the code. Keep in mind that if you load the code with Control Key depressed, you also have to depress the Control Key to enter the code when you run the program. You can also put this Run Protector anywhere in the program as a Gosub routine providing the very first line says Gosub and the line number of the routine.

If you do not want anyone to have access to your private programs, then use both of these ideas. The first method prevents anyone from LISTing a program, to gain access to the code or contents, and the second method prevents them from RUNning the program.

Next time - how to type a program without losing data -

Scott Copeland.

TI - RICHARD TWYNING'S MINDER!



This is the latest in our series of articles featuring ineresting or unorthodox uses that our readers have put their computer to.

You may remember Richard Twyning for his letter in the December issue offering ways of producing such vital sound effects as 'rattle snake', 'helicopter', 'steam train' and 'man sawing'. Richard's fertile imagination and flair for programs with purpose has moved him to send us such routines as "turning all the screen upside-down", and "disabling the keyboard". (By the way, Richard - I've lost my copies of those routines. Any chance of another cassette?)

Thus, if anyone had to come up with the following program, I guess it just HAD to be Richard:

Dear Sir,

I have read through the latest 99/4a magazine, and I read on page 23 that you wanted to know if anyone had any strange uses for their TI-99/4a. I have made my own interface which connects to the RS232 port. It uses two transistors to amplify the signal from the computer, so I can control up to six electric circuits.

e.g.

I open both RS232 channels, RS232 and RS232/2, and I have connected two LEDs (Light Emitting Diodes) so when I put PRINT 1, one LED comes on, and when I put PRINT 2 the other LED comes on.

I have also made some pressure pads out of two pieces of cardboard and some wire, and I have put a magnetic sensor on my bedroom door.

I have connected the pressure pads and the magnetic sensor to a joystick interface that I have made, and they can be detected by CALL KEY(1,K,S) and CALL KEY(2,K,S).

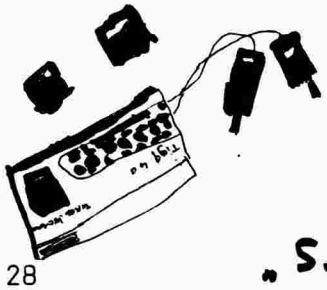
I have written a burglar alarm program and I have connected an electric siren to my RS232 interface, so that when the door opens the word DOOR is printed on the printer and the siren is switched on, and when anyone stands on the pressure pad that is next to the window the word WINDOW is printed and the siren will be switched on.

Richard Twyning

EDITOR'S FOOTNOTE:

Well, what can I say?

(I understand that Richard is now devoting his time to developing a program that measures earth subsidence under his house, and that his parents are developing paranoia.)



ANSWERS
KEYBOARD
PRINTER
TAPES
JOYSTICKS
SOUNDS
CARTRIDGES
GRAPHICS
VARNISH
MINI MEMORY

"LITTLE BITS"

cant do that

Thanks go to Richard Owen for offering to deal with problems through this page. He does not claim to know all the answers, but he will help where he can, and encourage others to chip in with their suggestions as well:

From time to time, people come across many different things that they don´t understand about or can´t do (hence the name)

This month´s first ´Can´t do that´ is from Mr. R. Fearn. He wants to know what the first digit in the EXTENDED BASIC, "CALL ERR() command is for.

Well Mr.Fearn,The first digit as far as we can see is for the error code, which we can only determine as in the APPENDIX N part of the EX-BASIC manual, where on page 217 it gives a list. These are in numeric order.

As for the numbers that aren´t mentioned, we can only guide, and as far as we can see, #2 refers to the joystick, and #3 to the keyboard.(This is only a guide!)

The next question is from Mrs. P. Hawes - actually she delivers two such questions, tne fist being: ´How do I go about converting one machine´s BASIC to another?´

The easist way, Mrs Hawes, is to buy a book from a well known newsagents in the high street, and go from there; but one word of warning: with only the basic system you are going to come across knotty parts to some programs which are unobtainable on the basic TI.

The second question asked is ´Is tnere any way around the ´peek´ and ´poke´ routines on other machines?´ The simple answer is No (on the basic machine), but Yes with either Minimem or Ed/Ass or Ex-basic & 32k ram.

The next one is from Mr.Tony Goodall, who asks ´Why is it that sometimes in EX-BASIC when entering data,it only allows two lines?

The answer is, Mr.Goodall, that due to the lack of memory available in EX-BASIC, it only allows certain data amounts. To avoid any mishaps, I´d keep it to about two and a half lines maximum.

Thankyou Richard, and if anyone else can shed further light on any of the above, please write to us.

T.I.TBITS

99/4a Magazine
Parco Electrics
2 Devonshire Court
Heathpark
Honiton
Devon

This column was introduced in the first volume of 99/4a and proved very popular. This is where to send those interesting routines and tips that maybe don't constitute a full feature or program, but which you think others will be interested to see.

I'll kick off with a quickie about CALL LOADs in EB with 32k.

CALL LOADs are a way of directly accessing memory locations of your console.

The format for using them is as follows:

* 100 CALL INIT
  110 CALL LOAD(-nnnnn,n)

* (CALL INIT should only be used once in the program.)

There are several being used by people these days, and what follows is a summary of some of the most popular and useful examples:

CALL LOAD(-31748,nn) where nn=1 255 - using this will speed up the flashing of the cursor. 1 is normal, and 255 almost invisible. A drawback is that it also increases the length of any tones generated. Try 10.

T.I.TBITS

CALL PEEK(2,A,B)
CALL LOAD(-31804,A,B) - Use this in place of a STOP statement in a program and it will automatically return you to the TI title screen.

CALL LOAD(-31806,nn)
when nn=16 Disables FCTN/QUIT
nn=32 Disables ERROR BEEP
nn=64 Disables SPRITES

Have fun!

Next, Tim Anderson, who proposes the following alterations to our 'BASICBALL' program to give greater control to the players:-

```
490 CALL SPRITE(#3,70,B,169,
121):: CALL DELSPRITE(#1)::
FOR N=1 TO 150 :: NEXT N ::
GOSUB 1320 :: CALL CHAR(70,A
$):: O=INT(RND*3):: CALL CHA
R(70,B$):: FOR N=1 TO 10 ::
GOSUB 1320
500 NEXT N :: CALL CHAR(97,D
$):: GOSUB 1320
505 CALL KEY(G,EE,DD):: IF E
E<>18 THEN 505 :: CALL CHAR(
97,E$,97,F$,97,G$):: CALL MO
TION(#14,0,0,#16,0,0,#17,0,0
)
510 CALL SOUND(100,-5,5):: C
ALL SPRITE(#10,58,5,145,122,
12,2,#1,58,16,145,122,12,2):
: CALL POSITION(#4,P,Q,#5,P,
R,#6,P,S)
520 CALL CHAR(97,C$):: T=1 :
: CALL JOYST(H,E,D):: IF D<>
0 THEN CALL CHAR(70,A$):: GO
TO 550
530 FOR U=1 TO 2 :: CALL JOY
ST(H,E,D):: IF D<>0 THEN 550
540 NEXT U :: CALL JOYST(H,E
,D):: IF D<>0 THEN CALL CHAR
(70,A$):: GOTO 550 ELSE 560
550 CALL CHAR(70,A$):: CALL
POSITION(#1,DB,DC):: IF DB>1
75 OR DB<164 THEN 560 ELSE C
ALL MOTION(#1,-26,3*(DB-170)
+O,#10,-7,3*(DB-170)+O)
551 CALL SOUND(1,-7,0):: M=0
 :: GOTO 580
560 CALL CHAR(70,A$):: CALL
DELSPRITE(#10,#1):: CALL SOU
ND(30,110,0):: M=M+1 :: DISP
LAY AT(1,16):" STRIKE";M ::
GOSUB 1320 :: IF M=3 THEN 57
0 ELSE 440
```

Thermal Paper Parson strikes again! No, that comment has nothing to do with the fact that he is a teacher, but it does indicate that he has been busy on the TI. He has sent us a program to demo XYZ graph plotting 'A' level maths. Now I'm sure we'd all like to see it - how about sending us a copy on tape, TP? Two reasons: a) Much of the printout (on Thermal Paper) is almost illegible, and b) I'm too darned idle to type it all out anyway.

George O'Sullivan writes to tell us that both TI WORD and MONKEY GRAPHICS in previous issues can be converted to run in TI Basic.

e.g. Line 150 in MONKEY GRAPHICS:

```
150 FOR I=3 to 8
151 CALL COLOR(I,I,1)
152 NEXT I
153 CALL COLOR(1,14,2)
154 CALL COLOR(2,16,1)
```

The 'Display At' and 'Accept At' commands can be substituted by 'Print' and 'Input' statements. Change 'On Error' commands to REMS.

If you have the PRK module the following can be used:

```
250 CALL D(6,1,16,"3. LOAD A
PROGRAM",7,1,14,"4. SAVE PRO
GRAM")
560 CALL A(17,10,1,FA,A$)
```

PS. Are there any TI Users in Ireland?

The unstoppable Stephen Meadows has written in again, this time he offers the sound of a church clock!

```
100 FOR T=1 TO 7
110 FOR I=0 TO 29
120 CALL SOUND(-1,444,I,110,
I,140,I,-1,I+1)
130 NEXT I
140 NEXT T
```

What about a car:

```
100 FOR S=30 TO 0 STEP -1
110 CALL SOUND(-99,-5,S,210-
S,S/2+1,160-S,S/2+4)
120 NEXT S
130 FOR I=0 TO 30
140 CALL SOUND(-99,-5,I,111+
I,I/2+1,130+I,I/2+1)
150 NEXT I
```

TI.TBITS

Here's an TItbit that has the effect of performing a NEW command, from within a program:

```
100 CALL CLEAR
110 ON ERROR 100
120 RUN ""
```

Stephen also spotted a spelling mistake in Parsec: ASTERIOD BELT! He says surely that should be ASTEROID?

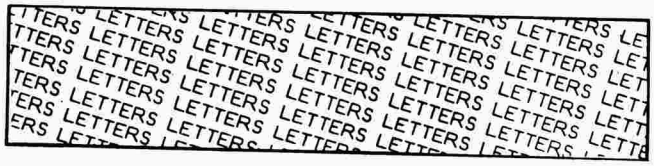
For those of you who want to transfer Adventures from tape to disk, and who also have Tunnels of Doom, here is a way that's pretty effective:

First, load in an adventure file using 'Tunnels' menu. Then save to disk (do this by pressing BACK after seeing the screen enquiring whether to continue or restart the game). Then plug in Adventure module and load the game from disk. (DSK1.Filename).

The game plays as normal. but with a few minor hitches such as redefined letters (not many) and the two arrows that appear at the ends of the separating line (I don't have a clue why!)

This, I know, is of limited use due to the number of disk users, but should make it more convenient for those with disk and 'Tunnels'.

Lastly, Stephen submitted a routine to free memory on startup with EB and disk-drive. The crucial factor involved was the use of the statement CALL FILES(1) within a program. While I am happy to publish the routine, I will first invite comment from Stephen or anyone else about it, since my console will not accept CALL FILES(1) as part of a program, only as a command in the immediate mode. Comments, anyone?



DEAR PARCO

HI,I ENCLOSE MY LATEST ORDER FOR YOUR QUALITY SOFT WARE.

I WANT TO WRITE TO YOUR MAG,SO LISTEN.

THE FIRST THING THAT HIT ME WAS THE PRICE 2.25 A LITTLE STEEP,WHERE COMPUTER,VIDEO GAMES IS 95p

THEN I THOUGHT ,THIS IS THE ONLY MAG THAT IS DEDICATED ENTIRELY TO THE TI,SO IT MUST BE WORTH IT.

LOOKING INSIDE p.2 WHAT IS THIS CRAP ABOUT CRAPO EJECT.

IS THIS ONLY FOR WIERD READERS? I DONT FOLLOW ONE WORD.

WHY HAS IT YOUR OLD ADRESS?

THIS MAG IS NICELY PRESENTED,YOUR OWN DRAWINGS ON THE COVER?. I LIKE THE LETTERS PAGE,BUT MAKE IT LONGER. NEWS PAGE IS OK. NOW FOR THE PROGS.

THEY ARE LONG ARE'NT THEY,I HOPE YET? THEY'RE BUG FREE.

I LIKE KEYBOARD KRUNCHERS PAGE, IT HELPS THE READER DECIDE IF THE PROG IS WORTH TYPING IN.

cont.. ... P23 IS BRILLIANT. THIS SHOWS THE TI AS A KIND OF SUPER HERO LIKE SUPERMAN FOR INSTANCE SAVING LIVES, INSTEAD OF BEING LOCKED UP AT HOME, PLAYING GAMES WITH THE KIDS.

I HOPE TO READ MORE CASES LIKE THIS,VERY INTERESTING.

BRIGHT SPARKS HMM. NOT MY CUP O'TEA I'M AFRAID''.

FOR THE MORE SERFISTICATED PROGRAMMAR I PRESUME.

NOW,THE REVIEWS PAGE,FANTASTIC,I LOVE THE REVIEWS PAGE.

WITH ALL THE CRUMMY TEXAS INST. MODULES LET LOOSE (WHITCH ARE YEARS OLD) WE NEED TO BE WARNED.

THIS PAGE(S) NEED TO BE DOUBLED TO 4.

I CANT DO THAT .. THIS PAGE IS GREAT IT SET ME THINKING . WHEN I TRY PROGRAMMING,IM FAMILIAR WITH THIS STATEMENT.

I COULD WRITE YOU A FEW I CANT DO THAT'S..ANOTHER TIME PERHAPS.. BOOK REVIEW ,OK PUT A TRIFLE BORING.

WHAT EXACTLY DO YOU WANT IN LITTLE BITS? CAN I HELP?

JOYSTICK BENDERS WHAT A PAGE, ALL THOSE CRAZY GAMERS

I WILL HAVE TO PUT MY SKATES ON IF I WANT A SCORE PUT UP.

GOOD TO SEE A TOP OF THE POPS TYPE O' THING.

NOW THE BACK COVER. BULLITIN BOARD . I AM VERY CONFUSED, RADIOHM? PTO PLEASE.

CONT..AGAIN..... CALL SIGN BLAH-BLAH BLAH

HEADACHE HEADACHE HEAD ACHE .... ETC..ETC...

CAN I USE THIS FOR SELLING SOFTWARE OR IF I NEED STUFF.

NOW IVE CONE THRU THE MAG PAGE BY PAGE LET ME THINK..

WHAT IS MISSING.. HMMMmmm&??

AH YES... YOU KNOW YOUR PRICE LIST, WHY NOT REVIEW YOUR CURRENT SOFT WARE...HARD WARE'???

TELL US WHAT HAPPENS ,IF ADDITONAL ?WARE IS NEEDED.

INCLUDE A LITTLE PICTURE OF THE SCREEN(S)

DRAW PICTURES STRAIGHT FROM THE TELLY/MONITOR.

YEAH WHAT A GREAT IDEA???

WHAT ELSE.....????????????????...

O' WELL I CAN'T THINC OF ANYTHING ELSE AT THIS TIME

PUT AS THE EXTERMINATOR ALWAYS SAYS (IF YER LYING)

((I'LL BE BACK''))

TILL I GET A ST SUB FOR YOUR MAG

SEE YAH

YOURS SINCERELY

P. Tompkins

MR P A TOMPKINS
341 KENTON LANE
HARROW,MIDDX,HA3 8RT.

***** JOYSTICK BENDERS *****

BUCK ROGERS	292,194
MOON MINE	101,990
MASH	4,077
BURGERTIME	500,000
HOPPER	31,250
STAR TREK	660,110
JAW BREAKER	6,435
PARSEC	10,842,800
ALPINER	92,186
TI INVADERS	54,121
CAR WARS	56,250
TOMBSTONE CITY	1,000,000
THE ATTACK	388,800
BLASTO	47,340
MUNCHMAN	204,000
CHISHOLM TRAIL	89,990
MUNCHMOBILE	27,290
PACMAN	43,180
MS PACMAN	151,610
MOON PATROL	133,150
JUNGLE HUNT	29,180
POLE POSITION	109,850
DEFENDER	104,150
DONKEY KONG	1,741,300
PROTECTOR II	125,764
PICNIC PARANOIA	1,750
FATHOM	5,881
SLYMOIDS	113,905
JEGGIT	2,635
SUPERDEMON ATTACK	3,425
MOONSWEEPER	19,150

***** TOP FIVES *****

Game Five

American Football
Star Trek
Car Wars
Big Foot
Burgertime

Tape Five

Ti Drone
Dragon Slayer
Parco Golf
Parco Pack 1
Virgin Fun Packs

Serious Five

TE 2
Extended Basic
PRK
Touch Typing
Artist

Bookworm Five

1 Dynamic Games
2 Mastering Theti
3 Compute! Ass- ng
4 Get Started
5 Texas Programme

99/4a Magazine
Parco Electrics
2 Devonshire Court
Heathpark
Honiton
Devon

PRINTED BY

PARCO Electrics