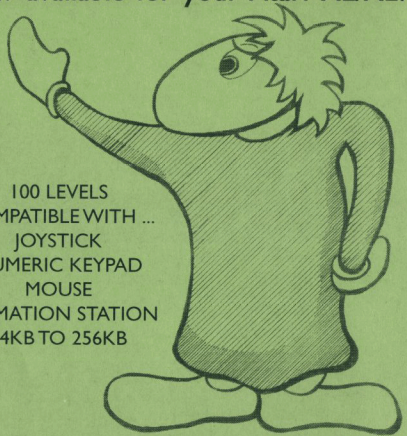
The Brundles

Now available for your Atari XL/XE!



100 LEVELS
COMPATIBLE WITH ...
JOYSTICK
NUMERIC KEYPAD
MOUSE
ANIMATION STATION
64KB TO 256KB

Ask your local software dealer or write to ...

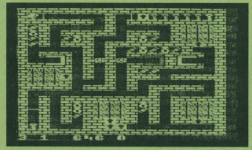
KE-SOFT

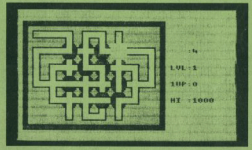
KE-SOFT KEMAL EZCAN
FRANKENSTRASSE 24
63477 MAINTAL - GERMANY

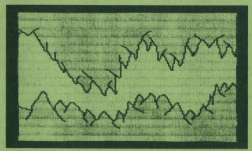
ZONG

Issue 1

ATARI XL/XE MAGAZINE + DISK







Featuring ...

News & Reviews
Cheats & Maps
Workshops
Software Disk
and lots more ...

DISK INCLUDED

PROGRAM-DISK

USING THE DISK

The ZONG disk is double sided medium density. It contains a DOS.SYS, AUTORUN.SYS (Turbobasic XL) and a menu.

Loading instructions: Turn off computer, turn on disk drive, insert disk either side, turn on computer. After a few seconds of loading a menu appears. Use cursor keys (do not press CONTROL) or Joystick to select programm, press RETURN or Trigger to load and run selected program.

See articles in magazine for program descriptions.

You can also use the menu for your own disks. Simply copy it onto a disk containing DOS.SYS and AUTORUN.SYS (Turbobasic XL). It shows and runs all .TB and .COM files

ZONG

ZONG ATARI XL/XE MAGAZINE + DISK is created by KE-SOFT Kemal Ezcan, Frankenstrasse 24, 6347 / Maintal, Germany. Phone +49 6181 87539, Fax +49 6181 83436.

A single issue is GBP 4.- / USD 6.50 / DM 10.-. ZONG can be ordered from your local Atari 8-Bit dealer or directly from KE-SOFT. Please send IRQ for pricelist.

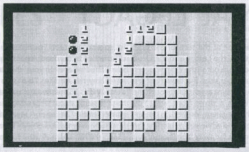
This is the first issue of an english ZONG magazine. If there's enough interest, there'll be lots more of it! ZONG exists in germany since 1989 as a printed magazine. Please tell your friends about ZONG and DON'T COPY IT!

Have lots of fun!

Kemal Ezcan

REVIEWS

Bomb



Owner's of MS-DOS PC's will know that game, it's the conversion of the Windows-game "Minesweeper". Object of the game is to find all hidden bombs on the playfield by klicking on desired pieces. If there's no bomb behind the chosen piece, the piece will be blanked or covered with a number indicating the number of bombs behind the 8 pieces around. By using this number you'll have to find (but not klick on!) all bombs. If done so, the level is cleared, you'll get a time-bonus and the next level appears. There are 20 levels at all which can be selected from the title screen. That's the first difference between "Minesweeper" and "Bomb". At "Minesweeper", each level is a rectangular playfield, it's size depending on the level. At "Bomb", each level has a different layout, there are ovals, figures, even a Pac-Man and a ghost!

The game features simple, but good-looking graphics, funny playfields, an easy to understand concept and hours of fun. While playing, a funky background tune plays. If you're looking for a new brainkiller which doesn't cost lots of money, you're right with "Bomb".

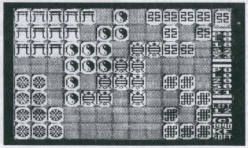
"Bomb" by KE-SOFT, Disk, GBP 6.50 / USD 10.00 / DM 14.80.

Ashido

In old China, people swear on the method of holy stone-meditation. Even today some followers of this old art can be found. The principles of this are easy to understand, but very difficult to master. With this game, you are able to peek into the famous chinese art of stone-meditation.

The game-board is divided into 12*18 squres. Each square can hold one stone. At the beginning, in each corner of the game board one stone, and in the middle two stones are placed. Goal of the game is to place all 72 stones on the game-board. There are six different stone-symbols with six different colors each, each stone appearing two times. But stones can only be placed beneath other stones which either match in color of symbol! It is not possible to place a stone without touching at least one which matches in color or shape! The more stones a placed stone touches, the more points you earn. If you get a "four-way", which has four stones around, score is doubled!

The game offers a choice of three different stone-sets. The first is for beginners and contains simple forms, the second is Chinese, the third Egyptic. Even during the game the stone-set can be switched.



As said before, the game is easy to understand, but difficult to master. It takes lots of practice to get some "four-ways", which is the only possibility to reach higher scores. The top-score will be saved to disk.

The screenshots shows one of the stone sets after some stones are placed. "Ashido" features a musical tune at the title screen and of course detailed multicolored high resolution graphics.

If you liked "Shanghai" or any other thinking game with chineese symbols on tiles, "Ashido" is the right one for you! There's no time-limit so you can lean back in your comfy chair (oh no, not the comfy chair!) and learn about chinese meditation ...

"Ashido" by KE-SOFT, Disk, GBP 8.50 / USD 13.50 / DM 19.80

FINAL LEGACY

Weigh anchor at the turn of the tide! You must sail the Legacy into the Dead Zone and knock out all the Warmongers' missile launcher sites. But beware of fire from enemy ships as you sail through the foam. This is a three-front war. You must fight the enemy on land, in the air, and at sea. And while you're battling the Warmongers far away from home, you must also protect your cities from destruction by the deadly enemy missiles - launched whenever your ship strikes an enemy missile site. Your ship must destroy 8 to 16 missile sites and battle 3 to 7 enemy ships, depending on the game level. On every level above ENSIGN - from LIEUTENANT through FLEET ADMIRAL - 4 to 14 sites are hidden from view. You can uncover them only by sinking enemy intelligence ships. You must defend 8 cities - NORK, BEGIN, LUND, BON, KYO, LIBERTY, KENDALL and WARN. Prepare for a long and lonely voyage!

During the game you can choose navigation and the battle mode from a menu. In sea-to-sea mode you have to defeat enemy ships with an "Silent Service"-like view, in sea-to-land you must protect your cities on a 3D-scrolling grid-landscape, in sea-to-air you eliminate incoming missiles, which looks a little like "Star Raiders".

The game features a good mix of action and strategy with nice graphics and sound effectes.

"Final Legacy" by Atari, Cartridge, GBP 10.50 / USD 16.50 / DM 24.80

STAR RAIDERS II

Star Raider Commanders are scarce in this galaxy. After wiping out the Zylon Empire most of them scattered and retired. But not you! You spent your time hyperwarping Federation Express Cruisers from one star system to the next. But now the Federation needs you - and we've got some real action!

Remember Celos IV, that peaceful star system? And it's planet, Teris (not Tetris!), a temperate paradise where battle-weary Star Raider Commanders went for rest and rehabilitation? Well, guess what! Teris is being invaded - by Zylons!

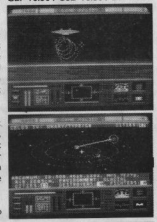
Some Zylon warriors who promised good behavior were freed to resettle on their home planet. But their upstart leader, Chut, preached revenge and changed their loyalties. Then he orchestrated a takeover of Zylon and the entire Procyon Star System. Chut now commands an awesome Zylon Master Force. And he's built Attack Bases capable of producing new Attack Squadrons in minutes. Zylon fighters are attacking Teris as we sit here!

We need action fast! The Federation is asking you to pilot the hottest new fighter in the galaxy - the Liberty Star. The mission is simple: Wipe out the entire Zylon Master Force. You must also penetrate their stronghold, the Procyon Star System, and destroy all their Attack Bases. If you don't, Zylon slaces will build enemy Squadrons as fast as you can eliminate them!

The battle zone covers two star systems: Celos IV, which you defend, and Procyon, where you attack. You must also protect the Federation Space Stations - your harbors for refuelling and repair - from being destroyed. Without the Space Stations, you're in serious trouble!

During the game you have to hyperwarp from planet to planet - or from squadron to squadron - destroying as many enemies as you can while protecting you Space Stations. The game features great graphics with very realistic animation of the space ships!

"Star Raiders II" by Atari, Cartridge, GBP 10.50 / USD 16.50 / DM 24.80



THE BRUNDLES

Yes, it's finished! After more than two years of working, you can play "Lemmings" on your Atari 8-Bit! And even more: It features not only the original-landscapes but some unique new ones like China, Ägypt and more!

You don't know "Lemmings"? Ok. Brundles (or Lemmings) are little mouse-like creatures who wander around without taking care of dangers. If there a fire in the way they get burned, if theres a hole in the ground they'll fall in. In each screen of the game the Brundles fall out a door and a certain number of them has to reach the exit to complete the level. What you do is to tell the Brundles what to do. This is done by clicking on one of the symbols in the lower part of the screen and then klicking on a Brundle. You can choose between Climbers (who can climb high walls), Floaters (who can fall a high distance without being hurt), Blockers (who stop others from passing a certain point), Bombers (who blow themselves up), Builders (who build bridges) and three kinds of Diggers (one digs straight down, one straight ahead and one diagonally down). With this features you have to lead the Brundles to the exit, but be careful: One Brundle doing the wrong thing and all can be lost! And there are up to 100 Brundles in each screen!

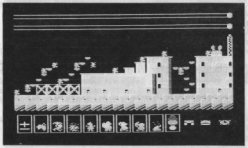
Before starting the game, you can configurate your computer system. The programmes recognizes a ramdisk and uses it. You can choose your input device from Joystick, CX-85 Numeric Keypad, Atari ST Mouse, Amiga Mouse and Suncom Animation Station! What a choice! Of course playing with a mouse is the best way. A mouse which comes complete with Atari 8-Bit software can be purchased from KE-SOFT.

The game features 100 levels each with it's own code, so you don't have to start again from the beginning. Just type in the code and start at the desired level. The 100 level, each consisting of more than one screen (smooth horizontal scrolling!) are made of 10 different landscapes including Woods, Caves, Cities, Volcano, China, Ägypt, Hell, Rome and more! Each landscape has it's own musical tune, so there are 10 tunes at all! Of course, the highscore can be saved to disk.

The first screens are easy to complete, so you just learn how to use the different features of the Brundles. Later the screens get real hard, you sometimes need more than one hour just to figure out what to do. Then, doing it the right way takes another one or two hours! One mistake and you see (and hear) all your lovely Brundles walking into a fire (then you have "Burn-dles") ...

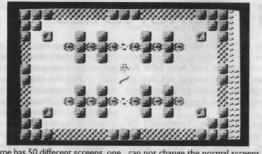
"The Brundles" is one of the best games ever written for the Atari 8-Bit, featuring all you want from a game: Great graphics, music, sound, lots of levels, variable control, highscoretable and ramdisk!

"The Brundles" by KE-SOFT, Disk, GBP 16.50 / USD 26.50 / DM 39.80



Lasermaze

Have you ever been a monk? In this game, you ARE one! Of course, no normal monk. The monk (who's name is actually "Monk"), has to hit all bombs in each level with his laser. The problem is that the laser can not be moved or directed, it shoot only in one direction. The only way to make the laser hit the bombs is to use reflector-blocks, which can be pushed around. So just put the blocks to the right positions, shoot the laser and hit the bombs. Sounds easy? Of course you have a limited number of shots, which can be refilled by hitting one of the ammunition symbols. The second (or third) problem are the killers. Hit one of them and you are killed.



The game has 50 different screens, one more difficult than the one before. It takes lots of time to complete all of them. Each level has it's own code so you won't have to start again from the beginning. With the included editor you can create your own screens, but you can not change the normal screens. "Lasermaze" features nice graphics and music and hours of thinking fun. But beware: It's not easy!

"Lasermaze" by KE-SOFT, Disk, GBP 8.50 / USD 13.50 / DM 19.80

Page Designer

The Page Designer can be used to layout a whole page which is divided in two halves, top and bottom. Both halves are in memory, each one the size of a graphics 8 screen. Pictures can be loaded and different character sets can be used. You can choose between 40 and 80-column textmode and there's also a built-in characterset with different shapes, lines and boxes. The draw-function allows to draw, usw lines, circles and balls in your page. The Page Designer is a very good tool to create simple layouts with not too much text, like disklabels, greeting cards and much more. The saved pages can easily be loaded into your own programs and therefore be used as title-screens. There are many different character-sets on disk. The Page Designer also works together with "Typesetter" to create a layout which can be loaded into Typesetter. Best combination are both programs! On the right you can see a page-designer printout (with german text).

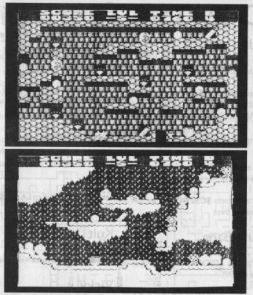
"Page Designer" by XLENT/KE-SOFT, Disk, GBP 10.50 / USD 16.50 / DM 24.80

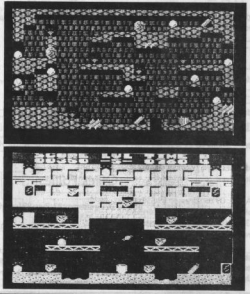
DONALD

Donald has to pay his rent! Therefore he accepts this dangerous mission. It is his task to travel to five different places and collect the famous golden eggs for his uncle.

In five levels, each consisting of 10 screens, Donald has to collect all golden eggs and put them into the teleporter which looks like a basket. You can choose with which level to start: The Amazonas with lots of snakes and spiders, Transsylvania with ghosts, bats and teleporter-mirrors. In the African Mines hidden caves make everything more difficult and at Himalaya offers icy paths, dark caves and blizzards! Last of all is the moon with aliens and spaceships.

Each level has different special items which can be collected for bonus. Dynamit can be used to break some walls.





The Moon

When Donald succesful wandered through all 40 other screens, the Moon can be selected! This is the most dangerous place of all. After a short stroll Donald finds a mysterious spaceship inhabited by monsters, eruptions, living floors, traps and labyrinths!

The graphics are comic-like nice and change in every level. Donald really walks like a duck. The seven different musical tunes are also very nice, also the sound-effects. The game can be played with Joystick or the CX-85 numeric keypad, which is the more easy way because of lots of jumping! Donald is a real great game featuring lots of levels, good graphics and sound. It takes lots of time to get through all levels.

"Donald" by KE-SOFT, Disk, GBP 8.50 / USD 13.50 / DM 19.80

CHEATS

Adax

Set adress $1A45 to xx = Energy

Set adresses $378F-37C1 to EA, $30E7-30E8 to EA, $30FB-30F9 to EA, $1DAB-1DAD to EA = invulnerable

Artefakt Pzrzdkow

Set adress $1EC3 to xx = number of lives (<=79)

Set adresses $C327-C329 to EA = unlimited lives

Bank Bang

Set adress $00D7 to xx = number of lives (<=99)

Set adresses $3C0F-3C10 to EA = unlimited lives

Captain Gather

Set adress $2984 to xx = number of lives

Set adresses $2B42-2B44 to EA = unlimited lives

Set adresses $35F1-35F3 to EA, set adresses $3600-3602 to EA = unlimited time

Change

Set adress $3BB9 to xx = number of lives

Set adresses $458F-4591 to EA = unlimited lives

Set adresses $3A46-3A48 to EA and $3A50-3A52 to EA = unlimited time

Dagobar

Set adress $3822 to xx = number of lives

Set adresses $4C64-4C66 to EA = unlimited lives

Set adresses $4E00-4E02 to EA and adresses $4E0F-4E11 to EA = unlimited time

Goldhunter

Set adress $00B0 to xx = number of lives

Set adresses $6BF6 to EA and $6C0B to EA = unlimited time

Kult

Set adress $4E78 to xx = number of lives

Vicky

Set adresses $95AD-95AE to EA = unlimited lives

Tac Tic

Set adress $006B to xx = number of lives

Set adress $00CB to xx = number of level

Set adress $0069 to xx = time (<6A)

Set adresses $9A39-9A3A to EA = unlimited lives

Set EA, A9, 09 to adresses $9C5E and the following, EA, A9, 59 to adresses $9C63 and following to get 10 minutes per level

All these cheats must be used with a freezer: Freeze the running programm and change the adresses.

Ms Pacman

Change Sector 104, Byte $1F from 04 to 09.

Pacman Jr.

Change Sector 122, Byte 2 from 04 to 09.

Treesurgeon

Change Sector 260 Byte $1E from 05 to FF.

Marauder II

Change Sector 228, Byte 19 from 04 to FF.

Basil The Great Mousedetective

Change Sector 128, Bytes $35, 36 and 37 from $CE, FD, 13 to $EA, EA, EA to get unlimited lives.

Quasimodo

Change Sector 63, Byte 3D from 04 to FF to get 255 lives.

Pac Man

Change Sector 33, Byte 48 from 03 to FF to get 255 lives.

Ollies Follies

Change Sector 82, Byte 99 from 05 to FF to get 255 lives.

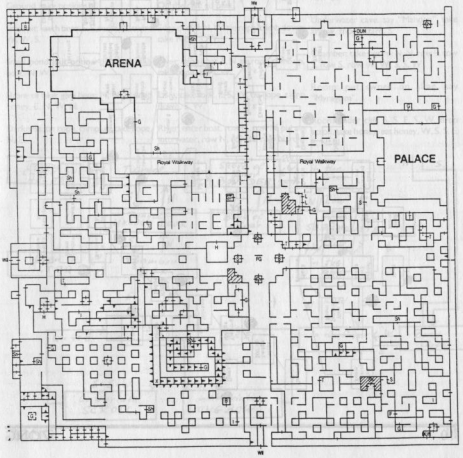
These cheats can be used with a diskmonitor. Just load the sector and change the bytes, then save the sector (use a backup!) and load the program.

Alternate Reality - The City

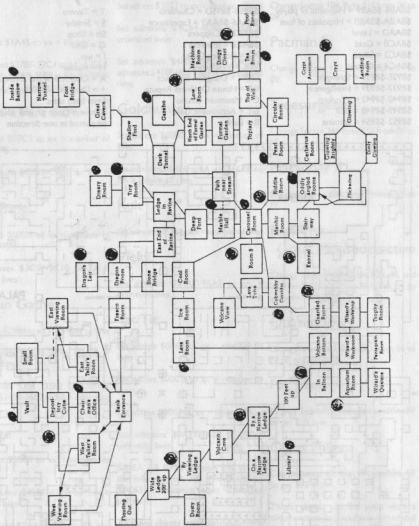
$8AB6-$8AB9 = Hitpoints of player
$8ABA-$8ABB = Hitpoints of foes
$8AAD = Level
$8AC0 = Gold
$8AC2 = Silver
$8AC4 = Gems
$8925-$8930 = Strength
$8937-$8939 = Intelligence
$8940-$8942 = Wisdom
$8949-$894B = Skill
$8952-$8954 = Stamina

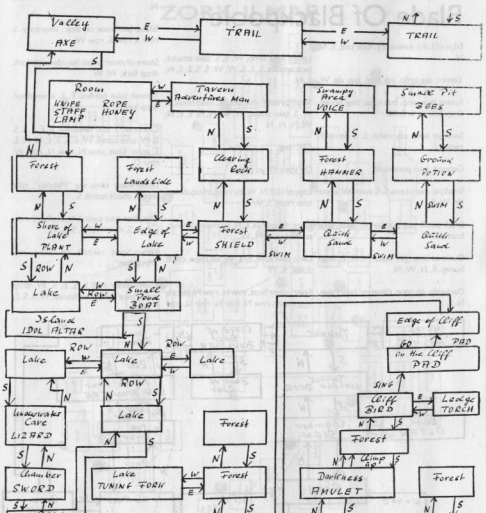
$895B-$895D = Charisma
$8AA6-$8AA9 = Experience
$8AC6 = Coppers
$8ACE = Jewelery
FG = Floating Gate
WIL = Wilderness
DUN = Dungeon Entrance
ILL = House Of Illness Repute
C = Maximum Casino
GT = Acriminiris Gate
B = Bank

T = Tavern
S = Smithy
Sh = Shop
G = Gild
I = Inn
H = Healer
X = Only follow the arrows
+ = Door or wall which can be passed from both sides
Arrow = Door or wall which can only be passed in one direction



ZORK II MAP


Pond
Inside Barrow
Mirror Room
Egg Room
Clearing
Canyon
Canyon
Landing Area
Clearing
Clearing
Clearing



NEW PD-DISKS

Superball II

A new version of the all time classic "Arkanoid". But this version is even better than the original one!

You have to clear the whole playfield of bricks with the ball hitting them. When hitting special bricks, extras fall down which are activated by catching them with your paddle. The extras are: Slow down ball, Wall under paddle, bigger paddle, laser, next level, magnet, extralives and superball. The superball breaks through all bricks without being reflected by them!

At the title screen you can choose between joystick, keyboard and paddle controllers. The game has really colorful graphics and lots of levels. Great value for money! Buy it, you'll love it.

Antic Database

Guess what: A database! First you have to format a disk and open a new file by choosing a name. Now enter the number of Fields, up to eight are possible. For each field you have to enter a name and length. the total length mustn't be over 128 bytes.

After this procedure, you can start to enter records. There's also a possibility to sort the whole file. Of course the program can't compete with professional ones like KE-BASE, but for normal uses it's enough and you'll easily get in touch with databases by using this program before trying a more complex one.

Warsaw Tetris

Another version of "Tetris", but one of the best. It was the first game from Poland which came over and actually it started the polish-games-boom. Different shaped bricks fall down the screen. The player has to flip and sort them to build full lines which disappear scoring points.

"Warsaw Tetris" has a nice title-screen, a top ten which is saved to disk, good graphics and nice music. If you don't have "Dredis", you should buy "Warsaw Tetris" (or Dredis, which is even better).

Jack The Digger

Everyone who knows the old game "Oils Well" knows exactly what kind of game this is.

A sort of "Pac Man", but with a pipeline. If one of the monsters touches the pipeline you loose a life, but you can catch the monsters when you touch with the mouth of the pipeline. The deeper you dig into the labyrinth the more difficult it becomes to get all points without being touched by a monster. The game is fast, making it even more difficult to complete a level.

If you like "Pac Man" type games and haven't got "Oils Well" in your diskbox, you should try this one, it's a really nice one.

Wizard Adventure Creator

If you want to create your own text-adventures, this program is the right one for you. The WAC is like a simple programming language. The first part is the editor where you enter all data like rooms, descriptions, items and so on. The compiler translates this data to a runnable basic-program which does not need any part of the WAC. That means you can play the created adventures without this disk!

Before you start creating an adventure with this program you should write it down, this is much easier! If you know a little about programming, it's even easy to include pictures in your adventure!

Computer Inhabitants Trouble With The Bubble

Two games on one disk. "Computer Inhabitants" is a conversion of the famouse "Little Computer People", where you can communicate with that little man living in your Atari, you can even play games with him.

"Trouble With The Bubble" is a conversion of "Bubble Ghost". You have to get a bubble through different screens by simply blowing at it avoiding lots of nasties. A very addictive game with good graphics.

DATA SORT

When using databases, lists or anything else with lots of diefferent entries the problem to sort the entries will occur. There are many ways to sort an array. A sort should be fast and, of course, without any bugs. The different ways to sort an array are called "Algorhythms". We'll explain three different ways which are demonstrated in the listing at the end of this article.

Bubblesort

It's a simple idea: Take the last entry and compare it with the one before. If the one before is smaller both entries will be swopped. The the last-but-one (supper) entry will be compared with the one before and so on. At the end of the first pass the biggest value is in the first entry. The second run does the same but stops before the first entry. To sort the whole array 20 passes (when using 20 entries) have to be made. It is called bubblesort because the bigger entries wander to the top.

Positive: Easy to understand way of programming. Negative: Up to 190 swops with only 20 entries, therefore not very fast.

Swop-Sort

We compare the first entry with all others. Is one entry bigger than the first, they will be swopped. After the first pass the biggest value is in the first entry. The second pass compares the second entry with all others and so on. After 19 passes the array is sorted.

Positive: Not so many swops as with Bubblesort, therefore faster. Negative: A little more complicated to program.

Memory-Swop-Sort

Instead of exchanging a bigger entry with the first, this method sets a counter to the bigger entry and compares that one with the others. After the first pass the counter points to the biggest entry which then can be exchanged with the first one. The program also need 19 passes but only a maximum of 20 swops!

Positive: The fastest way to sort. Negative: Nothing!

The Program

The program must be entered in TURBOBASIC XL. Lines 20 to 110 contain the mainroutine: First init some values, then input the entries for the array. After entering the 20 values you can choose between the three different ways to sort. The program shows the number of swops and the time needed to sort.

```
10 --
20 EXEC INIT
30 EXEC EINGEBEN
40 REPEAT
50 EXEC UEBERTRAG
60 EXEC ALGOWAHL
70 IF WAHL<>0
80 EXEC AUSGABE
90 ENDIF
100 UNTIL WAHL=0
110 END

120 --
130 PROC INIT
140 DIM EINGABE(20), FELD(20)
150 WAHL=0:TAUSCH=0:ZEIT=0
160 CLS
170 ENDPROC

180 --
190 PROC EINGEBEN
200 FOR SCHLEIFE=1 TO 20
210 ? SCHLEIFE;
220 INPUT "ENTRY: ";EINTRAG
230 EINGABE(SCHLEIFE)=EINTRAG
240 NEXT SCHLEIFE
250 ENDPROC

260 --
270 PROC UEBERTRAG
280 TAUSCH=0
290 FOR SCHLEIFE=1 TO 20
300 FELD(SCHLEIFE) = EINGABE(SCHLEIFE)
310 NEXT SCHLEIFE
320 ENDPROC

330 --
340 EXEC ALGOWAHL
350 REPEAT
360 ?
370 ? "0: END"
380 ? "1: BUBBLESORT"
390 ? "2: SWOP-SORT"
400 ? "3: MEMORY-SWOP"
410 ?
420 INPUT "YOUR CHOICE:";WAHL
430 UNTIL WAHL>=0 OR WAHL<=3
440 POKE 20,0:POKE 19,0
450 ON WAHL EXEC BUBBLE, AUSTAUSCH,AUSTAUSCHMERK
460 ZEIT=PEEK(19)*256+PEEK(20)
470 ENDPROC

480 --
490 PROC BUBBLE
500 FOR AUSSEN=1 TO 19
510 FOR INNEN=19 TO AUSSEN STEP -1
520 IF FELD(INNEN)<FELD (INNEN+1)
530 HILFE=FELD(INNEN)
540 FELD(INNEN)=FELD(INNEN+1)
550 FELD(INNEN+1)=HILFE
560 TAUSCH=TAUSCH+1
570 ENDIF
580 NEXT INNEN
590 NEXT AUSSEN
600 ENDPROC

610 --
620 PROC AUSTAUSCH
630 FOR AUSSEN=1 TO 19
640 FOR INNEN=AUSSEN+1 TO 20
650 IF FELD(INNEN)>FELD(AUSSEN)
660 HILFE=FELD(INNEN)
670 FELD(INNEN)=FELD(AUSSEN)
680 FELD(AUSSEN)=HILFE
690 TAUSCH=TAUSCH+1
700 ENDIF
710 NEXT INNEN
720 NEXT AUSSEN
730 ENDPROC

740 --
750 PROC AUSTAUSCHMERK
760 FOR AUSSEN=1 TO 19
770 ZEIGER=AUSSEN
780 FOR INNEN=AUSSEN+1 TO 20
790 IF FELD(INNEN)>FELD(ZEIGER)
800 ZEIGER=INNEN
810 ENDIF
820 NEXT INNEN
830 IF ZEIGER<>AUSSEN
840 HILFE=FELD(ZEIGER)
850 FELD(ZEIGER)=FELD(AUSSEN)
860 FELD(AUSSEN)=HILFE
870 TAUSCH=TAUSCH+1
880 ENDIF
890 NEXT AUSSEN
900 ENDPROC

910 --
920 PROC AUSGABE
930 CLS
940 FOR SCHLEIFE=1 TO 20
950 ? FELD(SCHLEIFE),
960 NEXT SCHLEIFE
970 ? :? "SWOPS: ";TAUSCH
980 ? :? "TIME (1/50 SEC): ";ZEIT
990 ?
1000 ENDPROC
1010 --
```

The different routines are easy to understand and can also be used in your own programs. If you have any comments, questions or maybe a program which can be published, just write a letter (or send a disk) to KE-SOFT, FRANKENSTRASE 24, 63477 MAINTAL, GERMANY.

PROGRAMMING GAMES

Everyone who always wanted to program a game and knows the basics of Basic and Turbobasic XL can learn how to program games. It's easier than it looks like!

Modules

We choose Turbobasic XL as the programming language because of it's speed and ease of programming. But we have to take care that our program will be as flexible as possible because we don't want to show you how to program a certain game but the basics of programming ANY game you want. Graphics, movement, animation and so on. So we'll put all things in Procedures and all values which maybe change later in Variables. Example: Instead of

```
GRAPHICS 1
COLOR 1: PLOT 0,0
```

to choose a graphics mode and plot a point, we'll use

```
EXEC GRAPHIKS_ON
EXEC SET_FIGURE
```

and

```
MODE=1
FIGURE=1
```

This way we can easily change the graphics mode oder the look of the figures (now only a point) later.

Movement

Now we start writing a program. First we want to create a figure which can be moved around via Joystick. Let's start with a mainroutine:

```
10 EXEC VARIABLES
20 EXEC GRAPHIKS_ON
30 EXEC GAME_START
40 DO
50 EXEC JOYSTICK
60 EXEC MOTION
70 LOOP
```

The procedure VARIABLES sets all important values for the game.

```
30000 --
30010 PROC VARIABLES
30020 MODE=3+16
30030 FIGURE=1
30990 ENDPROC
```

The procedure GRAPHIKS_ON turns on the graphics mode.

```
29000 --
29010 PROC GRAPHIKS_ON
29020 GRAPHICS MODE
29990 ENDPROC
```

The procedure GAME_START makes preperations for the game. At first, this in only the starting position of our player. It should start in the middle of the screen. To know the middle of the screen we have to know the size (resolution) of the graphics mode we use. Mode 19 (3+16, that means mode 3 without textwindow) has a horizontal resolution of 40 (0-39) and a vertical resolutuion of 24 (0-23). The middle of the screen therefore is X=19 and Y=11. But if want to change the graphics mode later the player should also appear in the middle, so the computer will take care of that.

```
30040 XMAX=39
30050 YMAX=23
```

Now it's very easy to get the middle of the screen:

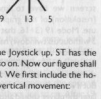
```
28000 --
28010 PROC GAME_START
28020 X=XMAX DIV 2
28030 Y=YMAX DIV 2
28990 ENDPROC
```

X is used for the horizontal position of the figure, Y for the vertical. The upper-left corner of the screen has the coordinates 0,0, the lower-right corner (in this mode) 39,23. X counts from left to right, Y from top to bottom.

After calling theese three procedures the procedures JOYSTICK and MOTION will be called endless (in a DO ... LOOP loop). This will move our figure around. Let's start with JOYSTICK:

```
27000 --
27010 PROC JOYSTICK
27020 ST=STICK(0)
27990 ENDPROC
```

If we call this procedure, we'll get the following values in the variable ST:



If we push the Joystick up, ST has the value 14 and so on. Now our figure shall move around. We first include the horizontal and vertical movement:

```
26000 --
26010 PROC MOTION
26100 IF ST=11
26110X=X-1
26120COLOR FIGUR
26130PLOT X,Y
26140 ENDIF

26200 IF ST=7
26210X=X+1
26220COLOR FIGURE
26230PLOT X,Y
26240 ENDIF

26300 IF ST=14
26310Y=Y-1
26320COLOR FIGURE
26330PLOT X,Y
26340 ENDIF

26400 IF ST=13
26410Y=Y+1
26420COLOR FIGURE
26430PLOT X,Y
26440 ENDIF

26990 ENDPROC
```

A note for everyone who already knows this: We also know that this can be done lots shorter, but the program shall be used for later changes!

We divided the listing to show the four ways of movement. We used greater line-number steps to have room for later inserts.

Now: If the Joystick is pushed in a certain direction (IF ST=), then the part between IF and ENDIF will be executed. If not, the program continues after the ENDIF. If the Joystick is pushed in a direction, the position of the figure will be changed and after that the figure will be plot again.

Now we can start the program for the first time. At first, nothing happens (Bug 1). If we move the Joystick, a figures (a dot) appears, but it draws a line instead of moving around (Bug 2). If we move out of the screen, an ERROR (141 or 3) occurs (Bug 3).

Let's check our bugs one after another. Bug 1: At first, nothing happens. This is because the figure will only be plot when the Joystick is moved. Let's insert:

```
28040 COLOR FIGUR
28050 PLOT X,Y
```

Now the figure will be positioned one time before the main routine. Bug 2: The figure draws a line instead of moving around. This is because we change the position of it without ereasing it before! This must happen in the procedure MOTION, four times. To make to program shorter, we'll use a new procedure FIGUREEREASE to erease the figure.

```
25000 --
25010 PROC FIGUREEREASE
25020 COLOR 0
25030 PLOT X,Y
25090 ENDPROC
```

This procedure has to be called by the procedure MOTION:

```
26105 EXEC FIGUREEREASE
26205 EXEC FIGUREEREASE
26305 EXEC FIGUREEREASE
26405 EXEC FIGUREEREASE
```

To make it even shorter we'll also do the setting of the figure in an extra procedure:

```
25100 --
25110 PROC FIGURESET
25120 COLOR FIGURE
25130 PLOT X,Y
25190 ENDPROC
```

Now we can delete some lines:

```
26130
26230
26330
26430
28050
```


and change some others:

```
26120 EXEC FIGURESET
26220 EXEC FIGURESET
26320 EXEC FIGURESET
26420 EXEC FIGURESET
28040 EXEC FIGURESET
```

A test-run shows: The figure moves fast and without drawing a line. Bug 3: The figure must not leave the screen. To do this, we have to use borders. We already have the varaibles XMAX and YMAX for the maximum values. We do the same with minimum values:

```
30060 XMIN=0
30070 YMIN=0
```

Of course this only isn't enough. We also have to tell the computer to accepe these borders. Only if the Joystick is pushed in a direction AND the figure has not reached the border, it can move.

Change line 26100 from

```
26100 IF ST=11
```

to

```
26100 IF ST=11 AND X>XMIN
```

The same with the other three directions:

```
26200 IF ST=7 AND X<XMAX

26300 IF ST=14 AND Y>YMIN

26400 IF ST=13 AND Y<YMAX
```

Now start the program. Yeah, it works!

The next thing we'll do is to recognice obstacles on screen.

Universal

At this point, our program has a small routine for each direction of movement. This is easy to understand. If we add obstacles now, the routine for each direction will be longer. The more obstacles, the longer the routine, four times each! So we now first do our movement in only ONE routine.

Let's take a look at lines 26100 to 26240. These are the directions left and right. What's the difference between left and right? Only the IF statement and the change of the variable X. At up and down it's the same, but the variable Y. So it should be no problem to use only one routine for all four directions.

The first difference, the IF statement, can not be changed. But the other one. We want to exchange the statements X=X-1, X=X+1, Y=Y-1, Y=Y+1 with one universal statement which does the same. This can be done by exchanging the values +1 and -1 with variables: X=X+XR instead of X=X+1. So we change the lines which change X and Y.

To get the right values in XR and YR, the procedure JOYSTICK has to set them.

```
27100 XR=0:YR=0
27200 IF ST=11 AND X>XMIN THEN XR=-1
27300 IF ST=7 AND X<XMAX THEN XR=1
27400 IF ST=14 AND Y>YMIN THEN YR=-1
27500 IF ST=13 AND Y<YMAX THEN YR=1
```

Line 27100 sets XR and YR to zero to ensure our player won't move around without stopping!

After inserting these lines, lines 26100 to 26440 are not useful any more. Let's delete them with

```
DEL 26100,26440
```

Now the procedure MOTION consists only of PROC and ENDPROC. Let's insert the movement, like bofre: First clear the figure, change the position then set the figure.

```
26100 EXEC FIGUREEREASE
26200 X=X+XR:Y=Y+YR
26300 EXEC FIGURESET
```

Line 26200 changes X and Y by the values of XR and YR. Start the program.

Flickering

If still works, but the figure now flickers. Why that? Let's take a look at the procedure MOTION. It first clears the figure, then changes the position and then set the figure again. But what if the Joystick isn't moved in any direction? Of course, XR and YR are set to zero, but the figure will still be ereased and set again. That's the flickering.

To stop this, we have to make sure the figure will only be ereased if XR or YR are NOT zero.

```
26050 IF XR<>0 OR YR<>0
26400 ENDIF
```

This IF ... ENDIF ecloses the whole movement. The figure will only be ereased if it also will be moved.

Obstacles

Now we have the movement in one short routine we can take a look at the next problem: Obstacles. Before we can recognize some, there have to BE some!

```
30080 WALL=2
```

We can set them in the procecdure GAME_START:

```
28100 COLOR WALL
28110 PLOT 0,0:DRAWTO 39,0
28120:DRAWTO 39,23:DRAWTO 0,23
28130 DRAWTO 0,0
```

These lines draw a border around the playfield. To also get some real obstacles

we'll draw to lines:

```
28140 PLOT 10,5:DRAWTO 10,18
28150 PLOT 29,5:DRAWTO 29,18
```

If we start the program now it look good, but the figure wanders through all walls and ereases them - that's not what we wanted!

Locate

To make the walls real walls which the figure can not pass, the figure must, before moving, check if there's a wall at the new position. The check what's at a certain position on screen we can use the LOCATE a,b,c statement. The first parameter is the horizontal, the second the vertical position. The computer uses the third parameter (which MUST be a variable) to write the value in that's found at that position. The statement

```
LOCATE X,Y,Z
```

for example writes the value thats at position X/Y in Z. We can use this in the procedure MOTION:

```
26060 LOCATE X+XR,Y+YR,Z
```

This statement write in Z what's in the position our figure wants to move to. Now the program only has to move the figure if there's no wall:

```
26090 IF Z<>WAND
26310 ENDIF
```

This IF ... ENDIF encloses the whole movement. So if there's a wall at the new position (which means the color the walls a drawn with), the figure can not move there. Try it.

If you're clever, you now think that the check if the figure is out of the screen is now useless, because there are walls at each side you can't pass. That's right, but maybe we'll remove the walls later.

Suggestion

Before trying this, save your program!

To get an interesting little game, you can draw a whole labyrinth and the figure has to find the exit. If you reach the exit, stop the program by pressing BREAK and check the values for X and Y (?X,Y). Then insert line 65:

```
65 IF X=0 AND Y=0 THEN END
```

Instead of the two zero insert the values you got just before. Now you can play the game and the program stops when you reach the exit.

Ok., enough playing, let's continue to develop the program. Load the saved program.

Items

Items must be collected. Before this is possible they have to be drawn. It's the same procedure as with the walls:

```
30090 ITEM=3
```

and

```
28200 COLOR ITEM
28210 PLOT 2,2:PLOT 10,2
28220 PLOT 37,2:PLOT 29,2
28230 PLOT 2,21:PLOT 10,21
28240 PLOT 37,21:PLOT 29,21
```

Start the program. We'll see eight blue dots which can be passed through. That's because we haven't located them yet. That's also very easy:

```
26350 IF Z=ITEM
26360 SOUND 0,100,10,10:PAUSE 2:SOUND 0,0,0,0
26380 ENDIF
```

Now when you collect an item a little soundeffect can be heard. It is just switching on a sound, waiting and switching it off again. We use this now to check the items are collected.

You Did It

Now collecting the items should make sense. The program should stop if all are collected. To do this, we'll just count the items. Because we know there are eight, we'll tell the program:

```
28250 NUMBER=8
```

This is not inserted in the procedure VARIABLES, because it's not a variable which always stays the same but changes depending on the number of items we plot.

Next step is to subtract one item each time we pass one:

```
26370 NUMBER=NUMBER-1
```

If we start the program and collect all items, then stop it and input

```
? ANZAHL
```

the computer responds with zero. This we must check. Lines 40 and 70 now are an endless loop. Because we want the program to stop if there are no more items, we change line 40:

```
40 REPEAT
```

and line 70:

```
70 UNTIL NUMBER=0
```

and insert:

```
80 STOP
```

so the program stops instead of continuing with the procedures (which would cause an ERROR).

PROBLEMS

Here you can tell us your programming problems. Just send a little letter and a disk with your program and describe your problem. The first three problems are from German users ...

ML-Routines

"My problem are Assembly-Language routines which make anything move or happen independent from the Basic program. I tried to use some routines from ZONG disks but didn't manage to get them working. I think the problem is that I don't have any experience with assembly-language. Although I tried to use this "simultaneousness" in my programs, but that always ended in very slow programs."

Solution: I think that's a problem lots of people have. Normally you don't need assembly-language routines to reach the effect you want. The benefit of assembly-routines is, as you mentioned, they do some things independent from the basic-program. But only routines which run during vertical blank interrupt do this! Not every assembly-routine is a vertical blank routine and therefore may stop the basic program when running.

To enclose such routines in your programs you have to know how they work, which part of memory they use and which parameters they need to work properly. If you know this it should not make any problem. Because you didn't mention which routines you tried, I can not tell you what you did wrong.

But to get what you want, you don't need assembly-routines! Look at the following program. It does two things, but not simultaneously. The program plots random dots on screen. If you press start it does a sound-effect.

```
10 GRAPHICS 19
20 DO
30 X=RAND(40)
40 Y=RAND(24)
50 C=RAND(4)
60 COLOR C
70 PLOT X,Y
80 IF PEEK(53279)=6
90 FOR I=1 TO 250
100 SOUND 0,I,10,10
110 NEXT I
120 SOUND 0,0,0,0
130 ENDIF
140 LOOP
```

When you start the program you surely notice that the sound effect stops the other part of the program. But that ain't neccesarily so: You can also create a sound effect during the plotting of the points. To do this, you just have to include the sound-effect in the main-loop and set a flag when START is pressed. Just change the following lines:

```
0 TH=256
75 SOUND 0,TH,10,10
76 TH=TH+(TH<256)
100 TH=0
```

Now delete lines 90, 110 and 120. Now both, sound and graphics, run simultaneously, but the sound-effect is much slower than before. To get the same sound effecte, change line 76 to

```
76 TH=TH+8*(TH<256)
```

This principle of programming can be used with every kind of technic, movement, sound and so on. If it becomes to slow, it's better to use assembly-routines.

Music

"My second problem is to include music in programs which runs in the background."

Solution: That's the same problem as before. We need exact details about which routines you used!

PM-Graphics

"Third problem: Fast movement of Player-Missile graphics in vertical direction."

Solution: You seem to know the basics of PM-graphics, so we just explain the vertical movement. Using Turbobasic, the "slower" moment of PM-graphics looks like the following example. Let's say the data of the figure you draw are in SHAPE$, length 16 bytes. When you want to show your player at a certain position, the following loop is neccesary:

```
FOR I=0 TO 15
P O K E
PMB+Y+I,PEEK(ADR(SHAPE$)+I)
NEXT I
```

PMB shall be the base adress of the PM-graphics. ADR(SHAPE$) is the adress of the string with the PM-data. Instead of that string you can also use READ A and POKE PMB+I+Y,A, then the data will be read from DATA-lines in your program.

To make this faster only the writing of the data has to be exchanged. Turbobasic has the MOVE-statement which copies a certain number of bytes from one memory location to another. The first parameter is the source adress, the second the destination adress and the third the number of bytes to be copied. Example: MOVE 10,50,20 copies 20 bytes from 10 to 29 to adress 50 to 79. Using this with your PM-graphics the FOR ... NEXT loop should be replaced by

```
MOVE ADR(SHAPE$),PMB+Y,16
```

Using this, the movement of the player is very fast.

Pictures

"My last problem: I want to include pictures drawn with Koala-Pad in my programs. I didn't manage that, I only got Design-Master (a german program) pictures, which are without color."

Solution: There are two differences between Design-Master and Koala-Painter files. The first, as you mentioned, the color. The second, and that's important, the Koala-Pictures are saved to disk in a compressed form. That means that the program does not simply save the 8-KB RAM of the picture to disk (which results in a 62-sector file as with Design-Master), but compresses the data to achieve a shorter file.

The first problem can be solved by choosing the correct graphics-mode, that's 31 for Koala. To load compressed pictures just use the assembly-routine presented in ZONG issue 7/92.

PROBLEMS?

If you have any problems with programming in Basic or Turbobasic XL, just write a letter to ZONG! We'll help you and publish your problem with a simple solution! If you have any games worth publishing, send them also!

Your ZONG-Team!

Tips

Adress 16

Memory-adress 16 is used to direct the interrupt sources of Pokey:

Bit 7: BREAK-Key
Bit 6: whole keyboard
Bit 5: serial dataintput
Bit 4: serial dataoutput
Bit 3: End of dataoutput
Bit 2: Timer 4
Bit 1: Timer 2
Bit 0: Timer 1

With POKE 16,64 and POKE 53774,64 the BREAK-key can be locked. 16 is the shadow-adress of 53774. With POKE 16,0 the whole keyboard can be locked.

Mega-Text

In some programs you may want to display very big letters on screen. When using normal text display, only graphics-modes 0, 1 and 2 are available, but even graphics 2 is too small. So what?

Use the TEXT-command of Turbobasic. This command allows to display text in pixelgraphics modes 3-11, 14 and 15. Depending on the graphics mode the letters are up to four times bigger than with graphics 2. Just enter the following lines:

```
GRAPHICS 5
COLOR 1
TEXT 0,0,"Hi!"
```

Now try mode 3 instead of 5 and it's getting even bigger.

Beside the possibility of displaying large text, the TEXT command also enables you to display upper- and lowercase characters, even inverse-video and graphics-symbols are possible. Different colors are displayed simply using the COLOR-command.

Presentation

When creating your home-videos you'll surely need some neat title-screens. The following little programm does this in an easy way:

```
10 GRAPHICS 7+16
20 SETCOLOR 0,11,8
30 COLOR 1
40 TEXT 28,44,"KE-SOFT shows"
50 PAUSE 100
60 --
70 GRAPHICS 5+16
80 SETCOLOR 0,4,6
90 TEXT 36,10,"a"
100 SETCOLOR 1,7,10
110 COLOR 2
120 TEXT 12,20,"K.Ezcan"
130 COLOR 1
140 TEXT 0,30,"production"
150 PAUSE 100
160 --
170 GRAPHICS 3+16
180 SETCOLOR 0,3,2
190 COLOR 1
200 TEXT 0, 0, "<CTRL F><3 *
CTRL M><CTRL G>"
210 TEXT 0, 8, "<CTRL V><3 *
SPACE><CTRL B>"
220 TEXT 0, 16, "<CTRL G><3 *
CTRL N><CTRL F>"
230 SETCOLOR 1,1,10
240 COLOR 2
250 TEXT 4,8,"ZONG"
260 PAUSE 200
```

When typing in lines 200 to 220 just type the characters indicated, for example press the CONTROL-key and "F".

As this program shows, it is very easy to create a nice presentation for home video-titling.

Experiment with the different possibilities of graphics-modes and the TEXT-command of Turbobasic!

THE DISK

The disk contains the following files ...

SIDE A

DOS.SYS	Disk Operating System
AUTORUN.SYS	Turbobasic XL
AUTORUN.BAS	Menu
BERGSHOO.TB	Bergshooting
CASTLES.COM	Runtime
AUTORUN.CTB	Castles Of Conf.
CASTLE1.CHR	Chset
CASTLE2.CHR	Chset
CASTLE3.CHR	Chset
CASTLE4.CHR	Chset
ML.ROUT.DAT	ML-Routine
SUICIDER.TB	Suicider
ZAUBERW.TB	Zauberwald
ZAUBERW.CHR	Chset
ZAUBER1.PFD	Screen
ZAUBER2.PFD	Screen
ZAUBER3.PFD	Screen
ZAUBER4.PFD	Screen
ZAUBER5.PFD	Screen
MOUSE.TB	Mouse
MOUSE.CHR	Chset
SABOTAGE.TB	Sabotage
SABOTAGE.CHR	Chset
INVENTI3.COM	Invention 13

SIDE B

DOS.SYS	Disk Operating System
AUTORUN.SYS	Runtime
AUTORUN.CTB	Shit Main Program
COMLOAD.DAT	ML-Routine
BILDA.PIC	Screen A
BILDB.PIC	Screen B
BILDC.PIC	Screen C
BILDD.PIC	Screen D
BILDE.PIC	Screen E
BILDF.PIC	Screen F
BILDG.PIC	Screen G
BILDH.PIC	Screen H
BILDI.PIC	Screen I
BILDJ.PIC	Screen J
BILDK.PIC	Screen K
BILDL.PIC	Screen L
BILDM.PIC	Screen M
BILDN.PIC	Screen N
BILDO.PIC	Screen O
BILDP.PIC	Screen P
BILDQ.PIC	Screen Q
BILDR.PIC	Screen R
BILDS.PIC	Screen S
BILDT.PIC	Screen T
BILDU.PIC	Screen U
BILDV.PIC	Screen V
BILDW.PIC	Screen W
BILDX.PIC	Screen X
BILDY.PIC	Screen Y
BILDZ.PIC	Screen Z

BERG SHOOTING

What a silly name, eh? Sure, that's german, it means something like "Mountain-Shooting", and that's what it is! Run the game and you'll see: There's a landscape with a shooting stand in front. Press START and the game begins. The crosshair moves downwards, you just have to press the fire button to shoot. If you wait too long, the crosshair moves on to the next target. You got only one shoot per target! If you miss more than three of the ten targets, the game is over. If not, you'll proceed to the next level, that's of course more difficult to master. Before starting the game, you can choose a difficulty-level be pressing SELECT. But don't try a harder level at first, it's even hard enough if you choose the easiest ...

SHIT

You will encounter this if you play this game!

Once upon a time a lonely fly lived it's live until a fat human came and ate her up! Lucky it was, it lived and so it tried to find a way out.

Ok, now it's your turn, guide the fly through the stomach and the whole body to reach the exit - shit! If you touch a wall you will die! And hurry up, or you'll be a dead parrot, sorry - fly!

The game uses hires pictures to create the playfields, you can simply change them by just drawing in them (use Atariartist or any other compatible program which can load and save compressed pictures). PS.: Boot B-side of disk with OPTION key pressed!

CASTLES OF CONFUSION

You want more money! That's the only reason you entered that old castle. Of course there are lots of dangers waiting for you!

After the title screens appears you can choose a difficulty (A, B or C) by pressing SELECT. Press START to begin the game.

Object is to get all the treasures and then leave the room through the door of course after getting the key! Your man can be moved left or right via joystick. To get to a floor above you, stand under a teleporter (looks like a shower) und push the joystick up.

Here are some of the dangers waiting ...

- Nails on the floor!
- Deadly acid!
- Crawling spiders and their webs!
- Floors disappearing under your feet!
- Fire!
- Ugly monsters!
- Disappearing Bridges!
- Time!

INVENTION NO 13

This is a musical tune by Johann Sebastian Bach. No, he's not the programmer but the composer!

Just enjoy this masterpiece of classical music. It runs during vertikal blank interrupt.

ZAUBER WALD II

Again such a silly name, and again it's german. It means "Magic Forest". You guide the daring wizard Marc in the secret forest of Thor and try to throw magic balls on the treasure. You'd rather take the treasure instead of throwing balls on it? Well, play "Castles Of Confusion"!

Ok, you play this game! Let's continue: A very silly monster lives in the forest, trying to eat you! You can also throw magic balls on the monster, but it won't die, it only stops moving for a few seconds. Another possibility is to hide behind a tree, but time is running, listen to the background sound which gets higher and higher!

And the next problem: The treasure is hidden in a cave! To get into the cave you have to throw magic balls to the walls and check out which walls you can destroy. But be careful: You have a limit of 100 balls per screen!

MOUSE

You are a mouse? No, you're not a mouse, but you help the mouse by moving around the blocks which form a track for the mouse to get the cheese! The mouse doesn't stop moving so you have to hurry, and think careful before you move the blocks! When the mouse can not move anymore it dies (what a silly mouse!). If you manage to let the mouse eat all of the cheese, you'll proceed to the next level. Hint: If the mouse is on a block, it can be moved together with it! Holding the fire-button speeds the mouse up, earning more points (and danger).

SABOTAGE

The decision is made! The secret headquarters of Dr. Barnuse have to be destroyed! The time-bome is placed but Dr. Barnuse has sent out his Robot-Towers to protect him. So again you are the only one who can reach the bomb, activate it and blow the doctor in 1000 pieces!

The secret headquarter is divided into six rooms, each containing a cannon against intruders and 10 Robot-Towers. the Robo-Towers can be shot, but if the canon is activated, new Robo-Towers appear. The canon has a protection-shield which only disappears when the canon shoots. The canon can be destroyed when the shield is inactive, but if there are at least five Robo-Towers the canon reappears after a few seconds!

If you eliminate all Robo-Towers AND the canon, a door opens to the next room. At the last room you have to shoot the bomb to activate it. The countdown starts at 99 seconds. In this time you have to get through all six rooms and out of the building or you will also be blown! Be careful: The last room opens only when all Robo-Towers and the canon are shot!

SUICIDER

Again one of those crazy money-seekers enters some old mines. Of course there are lots of dangers, like always. The dynamite can be used to blow up rocks.

Move the man via joystick, jump by pushing the joystick diagonally and blow dynamite by pushing the trigger while moving the stick to the left or right.

If time runs out, the man dies. Waterflasks refresh him!