

1987 / 2

KLUB  
MIKROELEKTRONIKY



ATARI®

Z P R A V O D A J

OLOMOUC

## Turbo BASIC - interpret

Rozsáhlý a velice dobrý extrémně rychlý interpret vhodný pro ATARI 800XL a 130XE. Vhodná je disková jednotka. Tato verze je vhodná pro strukturované programování.

V následujícím popisu znamená:

aexp - aritmetický výraz

sexp - řetězec/textový výraz A\$, "TEXT", CHR\$( ), STR\$( ),  
HEX\$( ).

lineno - číslo řádku

**>>....<<**jeden nebo více příkazů. V příkazu IF THEN se jedná o text za THEN. Jinak nejsou žádná omezení.

**>><<**může obsahovat i více programových řádků anebo část programového řádku.

Ve standartním ATARI Basicu jsou pro vytváření smyček k dispozici jen příkazy skoku orientované na čísla řádků a příkaz **>>FOR-TO-NEXT<<**. V Turbo Basicu jsou k dispozici další strukturální prvky vycházející z progr. jazyka PASCAL.:

IF aexp THEN lineno

IF aexp ENDIF

IF THEN (normální)

IF aexp THEN

IF aexp ELSE ... ENDIF - když je výsledek podmínky ( $\neq 0$ ) je prováděn program mezi IF a ELSE, jinak mezi ELSE a ENDIF. ELSE může být vypuštěno. Jako oddělovací znaménko po aexp neslouží THEN, ale (:) nebo ukončení řádku RETURN. Také před a za ELSE A ENDIF musí být stejná oddělovací značka. Po ELSE nesmí být použito číslo řádku. V Atari Basicu je pro takovéto podmínky používáno více příkazů GO TO s podmínkou.

REPEAT .... UNTIL aexp

opakuje příkaz **>>...<<** dokud není splněna podmínka. Testování je zde prováděno teprve na konci smyčky a proto je programová část mezi REPEAT a UNTIL provedeno minimálně jednou.

WHILE aexp ... WEND

opakuje **>>...<<** tak dlouho dokud je splněna podmínka. To znamená, že když není na začátku podmínka splněna neprovede se ani jednou.

### DO ... LOOP

Jedná se o nekonečnou smyčku. Příkaz **>>...<<** se provádí stále znova.

### EXIT

Příkaz slouží k ukončení smyčky a ke skoku na její konec. Tento příkaz se dá použít ve smyčkách **>>DO .. LOOP<<**, **>>REPEAT .. UNTIL<<**, **>>WHILE .. WEND<<** a také **>>FOR .. NEXT<<**. Tento příkaz představuje nouzový východ ze smyček a pro smyčku **>>DO..LOOP<<** je to dokonce jediný možný výstup povolený pro strukturované programování. Nouzově je možno smyčku opustit příkazem **>>POP:GO TO<<** lineno. Tento způsob nepatří do prostředků strukturovaného programování !!!

**\*F**

**\*F +**

Po tomto příkazu jsou smyčky **>>FOR NEXT<<** zpracovány odlišně. Před prvním průchodem je zkoušeno zda čítač dosáhl konečné hodnoty. K tomu jeden příklad :

**FOR I=2 TO 1: ? I : NEXT I**

1. počáteční hodnota je 2
2. čítač je zvýšen o 1
3. srovnání s konečnou hodnotou 1
4. opuštění smyčky, protože podmínka je splněna

Po **F +** je nejdříve proměnná I naplněna počáteční hodnotou 2 a potom je srovnávána s proměnnou I (konečná hodnota=1). Návazně na to proběhne skok na příkaz **NEXT I**. Neproběhne žádný výstup na obrazovku. Příkaz **>>F+<<** šetří často speciální dotazovací rutinu.

**\*F-**

Nastavuje opět normální způsob zpracování smyčky **>>FOR..NEXT<<**. Program proběhne smyčkou vždy nejméně jednou. Vždy po spuštění programu příkazem **RUN** je automaticky prováděn způsob **>>F-<<** (odpovídá normálnímu Basicu).

**PROC name**

Začátek programu PROCEDURA s názvem name.

**ENDPROC**

Konec procedury (podprogramu)-odpovídá příkazu **GOSUB** lineno.

### EXEC name

Příkaz volá podprogram (proceduru) name - odpovídá příkazu GOSUB lineno. Normální příkazy GOSUB a RETURN dovolují volání podprogramů jen pod číslem řádku. Tento příkaz umožňuje volání podprogramů pod lehce zapamatovatelným názvem. Kromě toho zabírá příkaz **>>EXEC .. PROC .. ENDPROC<<** méně paměti než GOSUB - RETURN a je i o něco rychlejší. Jména procedur jsou, jsou ukládána stejně jako jména proměnných. Každé jméno zabírá 8 Byte+ jeden Byte na každý znak. Každé další použití zabírá jen jeden nebo dva Byte. Číslo řádku pro GOSUB příkaz zabírá při každém použití 7 Byte. Protože TURBO Basic je slučitelný s normálním Basicem je možno použít místo 128 až 256 různých jmen proměnných nebo procedur. Každý název nad 129 proměnných zabírá potom dva místo 1 Byte.

ON aexp EXEC pname,pname,...

odpovídá Atari Basicu příkazu **>>ON GOSUB<<**

pname = PROCedurname = jméno podprogramu(procedury)

#name, GO#name, ON aexp GO#nam,nam,nam

TRAP # name

RESTORE # name

Definice adresy **#** odpovídá příkazu PROC. Tímto způsobem se dají nestrukturované příkazy skoku GO TO udělat přehlednějším tím způsobem, že se příkaz GOTO číslo řádku nahradí příkazem GOTO # name. Také u příkazu TRAP a RESTORE je možno použít adresy(name). Proto je nutno použít znaku **#**, aby se interpreteru sdělilo, že nenásleduje číslo řádku, ale "adresa". Příkaz GO # name je rychlejší než GOTO řádek. Znak adresy **#**, který není u příkazu EXEC, PROC je používán proto, aby se zvýšila přehlednost a čitelnost programů.

POP

Tento příkaz platí pro programy s GOSUB, EXEC rovněž jako pro smyčky **>>FOR..NEXT<<**, **>>REPEAT..UNTIL<<**, **>>WHILE .. END WHILE<<** a **>>LOOP..ENDLOOP<<**. V podprogramu zabírá GOSUB, EXEC, REPEAT, WHILE, LOOP vždy 4 Byte(jako GOSUB ve standartním basicu). FOR NEXT zabírá však 13 míst, 12 Byte a to proto, že je nyní k dispozici 256 názvu proměnných a procedur. Ve stacku nejsou ukládána čísla řádků, což je jednou z příčin pomalého chodu Atari Basicu, ale adresy paměti těchto řádků. Provádění smyček u Turbo Basicu je proto rychlejší a

rychlosť neklesá se vzdáleností od počátku programu. Dále má ještě jednu výhodu. Pokud dojde k přerušení chodu programu právě chybu, dá se program editovat (opravit) a potom spustit příkazem CONT nebo GOTO. Ani proměnné ani Stack nejsou přitom vynulovány. To umožňuje psát programy, které si samy generují datové řádky nebo maží nepotřebné programové díly. (Příkaz POKE 842,13 vyvolá stav odpovídající stisku RETURN, POKE 842,12 nastavuje normální stav). K vymazání nepotřebných řádků slouží v Turbo Basicu příkaz DEL.

----

Specielní příkaz pro páznámku (REM). Při psaní programového řádku je všechno za tímto znakem ignorováno. Při LISTu je vytištěno 30 těchto značek. Tento příkaz zabere o 1 Byte méně paměti než příkaz REM bez textu. Pro odpovídající řádek ve standart Basicu je zapotřebí více paměti.

#### LIST

Při LISTování programů jsou smyčky opticky zvýrazňovány vsunutím dvou prázdných míst a tím jsou programy přehlednější. Kromě toho se tímto způsobem dají odhalit mnohé chyby jako více NEXT k jednomu FOR, více ENDPROC k jednomu PROC. Při nečistém programování je sice LISTING stejně jako interpreter ovšem v nepořádku. Takové konstrukce, které jsou v Atari Basicu jakož i v jiných verzích Basicu obtížně popsatelné se mohou v Turbo Basicu lehce nahradit příkazem **>>IF--ELSE-ENDIF<<** nebo EXIT. Z toho vyplývá opět snadněji čitelné programy.

#### \*L-

Vypíná tabulování. To může být nutné při editování dlouhých programových řádků nebo pro úsporu místa při ukládání na disketu (LIST"D:X"). Oddělovací řádek -- se po \*L- vytiskne jako dvojité místo třicetinásobné.

#### \*L

#### \*L+

Zapíná opět tabulování (normální stav po natažení interpretu). Dále je ještě možný list od čísla řádku do konce programu. LIST 3000 tiskne od čísla řádku 3000 do konce programu. LIST"P:",3000, odpovídá tisku na tiskárnu. , za číslem řádku dává počítači tisknout až do nejvyššího možného čísla řádku 32 767.

Nová chybová hlášení :

ERROR 22 ? NEST

Chyba řetězení - nastane když k příkazu WHILE není nalezen odpovídající ENDWHILE nebo ENDIF k příkazu IF nebo také po F+ příkaz NEXT k příkazu FOR. Při opuštění podprogramu pomocí RETURN nebo ENDPROC jsou smyčky přerušeny. To platí také pro smyčku FOR..NEXT i pro ostatní v Turbo Basicu použitelné smyčky.

ERROR 16 ? GOSUB - k příkazu GOSUB chybí RETURN

ERROR 13 ? FOR - k příkazu NEXT chybí FOR

ERROR 23 ? WHILE - k příkazu WEND chybí WHILE

ERROR 24 ? REPEAT - k příkazu UNTIL chybí REPEAT

ERROR 25 ? DO - k příkazu LOOP chybí DO

ERROR 28 ? EXEC - k příkazu ENDPROC chybí EXEC

ERROR 29 ? PROC - byla volána neznámá procedura

ERROR 30 ? # - byla použita neznáná adresa

ERROR 27 X PROC - (EXECUTING PROC). Toto chybové hlášení nastane, když je proveden příkaz PROC. Procedury mohou být volány pouze EXEC

ERROR 26 ? EXIT - EXIT bez smyčky

ERROR 15 ? DEL - Bylo vymazáno GOSUB k příkazu RETURN, NEXT k FOR, REPEAT k UNTIL. V Atari Turbo Basicu se dají editovat programy bez poškození hodnot proměnných nebo Stacku. Když se potom při návratu z podprogramu nebo smyčky narazí na odpovídající řádek, který byl vymazán nebo změněn, nastane

tato chyba. Tato chyba nastane je-li provedeno zabudované DEL v programu.

Všechna chybová hlášení standartního Basicu tištěná bez textu jsou v Turbo Basicu doplněna krátkým textem.

Př.: 138 TIME OUP, 29 PROC ...

Další příkazy :

DEL od, do

Vymaže programové řádky od, do ( včetne )

RENUM starý, nový, přírustek

Přečísluje všechny programové řádky od čísla starý. Nové číslování začíná číslem nový a zvyšuje se o hodnotu přírus-

tek. Všechna čísla před starý zůstávají mezměněna. Příkaz mění všechna čísla k příkazům GOTO,GOSUB,TRAP,RESTORE,LIST, DEL,ON GOTO a ON GOSUB. Při použití nedefinovaných čísel řádků je nastaveno odpovídající záporné číslo (př.: GOTO -100). Při vypočítaných skocích (GOTO var.,GOSUB 100+A.10, RESTORE A.10+1000) není přečíslování respektováno !?! Když je za příkazem číslo je toto číslo uvažováno, ale vše co je za ním zůstává nezměněno. Viz př. GOTO 1000+A.10 je číslo 1000 zpracováno stejně jako by byl použit jen příkaz GOTO 1000 zbytek je nezměněn. Když nenásleduje číslo, ale jméno proměnné nebo závorka, je příkaz zcela nezměněn.

DUMP

DUMP filespec

Tento příkaz vyvolá LIST použitych proměnných. Stejně jako normální LIST může být provedeno na tiskárnu DUMP"P:"

Příklad.:

A= 100 numerická proměnná  
B=10,1 pole,DIM B(9) nebo DIM B(9,0)  
C(0,0) nedimenzované pole.U pokí mohou obě možné dimenze být znázorněny zvýšené o 1  
D(10,10) DIM D(9,9)  
E\$ 10,20 String LEN=10,DIM E\$(20)  
F\$ 0,0 nedimenzovaný string  
G\$ 0,10 DIM G\$(10), LEN G\$=0  
H PROC 100 PROC H v řádku 100  
I #,120 Adresa I v řádku 120  
J ? Nedefinovaná adresa nebo PROC

Tisk proměnných a adres probíhá v tom pořadí, jak jsou uloženy v tab. proměnných.

TRACE

TRACE 1

Zapíná mod TRACE. To znamená, že se na obrazovce tisknou čísla řádků tak jak je program prováděn.

TRACE -

Vypíná se mod TRACE. Automaticky dojde k vypnutí tohoto modu, když dojde k chybovému hlášení. Čísla řádků PROC nebo # nejsou při vyvolání pomocí EXEC nebo GO # tištěny.

\*B

\*B+

Po tomto příkazu jestisknutí tlačítka BREAK zpracováno jako každá jiná chyba. Přerušení programu se potom dá zachytit pomocí příkazu TRAP a program je tak chráněn proti nechtěnému přerušení

\*B-

Vypíná výše uvedený mod. Při RUN je automaticky nastavem mod \*B-.

### Příkazy

DPOKE adr,slovo

DPOKE adr,slovo

Příkaz pro POKE pro slovo dvojité délky odpovídá příkazu:  
POKE adr,slovo-256.INT(slovo)/256):POKE adr+1, INT (slovo/256)

MOVE zdroj, určení, počet

Přesun bloku paměti odpovídá příkazu:

FOR I=0 TO počet -1:POKE určení+I,PEEK(zdroj+I):NEXT I

Pomocí příkazu MOVE 57344,NEWCHARSET,1024 se nechá překopírovat generátor znaků.

-MOVE,zdroj,určení,počet

Blokový přesun k vyšším adresám v případě, že máme málo volné paměti. Když by se bloky překrývaly, to je když platí zdroj+počet určení. Tento způsob zabrání poškození obsahu paměti. Příkaz odpovídá následujícímu programu :

FOR I= počet -1 TO 0 STEP -1: POKE určení+I, PEEK(zdroj+I):  
NEXT I

Příklad:Popsání celé textové obrazové paměti prázdným inverzním zankem MOVE DPEEK(88)+1,959. Neprovádí se žádné zbytečné příkazy pro uložení a je to rychlejší než smyčka v Basicu.

BPUT #n,adresa,délka

Psaní bloku.Výstup bloku.Odpovídá následujícímu programu:

FOR I=0 TO délka-1:PUT #n,PEEK(adr+I):NEXT I

BGET #n,adresa,délka

Čtení bloku.Vstup bloku.Odpovídá násled. programu:

FOR I=0 TO délka-1:GET #n,A:POKE adr+I,A:NEXT I

Pomocí těchto příkazů se dá ukládat a plnit paměť s maximální rychlostí.

Příklady.:

OPEN #1,8,0,"D:BILD.PIC":BPUT #1,DPEEK(88),7680:CLOSE #1

OPEN #1,4,0,"D:BILD.PIC":BGET #1,DPEEK(88),7680:CLOSE #1

Uvedené příkazy slouží k naplnění nebo uložení obrazové paměti na disk nebo z disku. Platí pro GR. mody 8,9,10,11,15. Pro ostatní mody se musí číslo 7680 upravit. Jinak by se nahrávaly nebo ukládaly nepotřebné oblasti, které nepatří k obrazu.

FILTO x,y

Těsnopisná zkratka, rychlejší a přehlednější než :

POSITION x,y:XIO 18#6,0,0,"S:" FCOLOR n

Volba barvy pro FILTO. Ve standartním Basicu je tento příkaz POKE 765,n.

CLS

CLS #6

Mazání obrazovky. Odpovídá A=PEEK 766:POKE766,0:POSITION 0,0:  
?(#6;)CHR\$(125);:POKE 766,A

PUT n

Odpovídá ? CHR\$(n). Například PUT 253 odpovídá ? CHR\$(253) Při PUT,GET,INPUT je také možné zadat >>#0<<. Použití >>IOCB 0<< s příkazem >>CLOSE<< nebo >>OPEN #0<<zablokuje řádnou práci obrazovkového editoru popřípadě stisknutí systém RESETU. Když u PUT chybí zadání # je automaticky použito >>IOCB#0<<.

GET KEY

Odpovídá OPEN#7,4,0"K":GET #7,KEY:CLOSE#7.

Čeká na stisk klávesy a přiřadí proměnné ATASCII hodnotu stisknuté klávesy. Jméno proměnné je libovolně volitelné

DIM

Při příkazu DIM jsou pole a stringy automaticky vynulovány, to je nastavený na nulovou hodnotu.

DIM A(100) odpovídá DIM A(100):FOR I=0 TO 100: A(I)=0:NEXT I

INPUT "text",var,var,...

INPUT "text";var,var

tento příkaz spoří dosud nutné příkazy pro print. Pokud je za textem středník, vytiskne se ještě jeden otazník. Pokud se napiše INPUT "";A je potlačen tisk otazníku.

TEXT x,ysexp

Píše text v grafické obrazovce, do místa x,y nastaví levý horní roh prvního znaku tištěného řetězce. (Počítá se v obrazových bodech!)

Příklad.: GRAPHICS 8:TEXT 50,90,"TURBO BASIC":TEXT 70,95,1000  
Na rozdíl od normálního příkazu PRINT, musí být po příkazu TEXT pouze tisk, nemůže být použit seznam s čárkou nebo středníkem. Kromě toho se tisk textu na konci řádku přeruší a není také možný SCROLING.

CIRCLE x0,y0,r

CIRCLE x0,y0,xr,yr

Kreslí kružnici kolem středu x0,y0 o poloměru r. Pokud se udají dva poloměry pro směr x,y nakreslí elipsu.

PAINT x,y

Vybarvuje uzavřený obrazec. Například kruh barvou udanou příkazem COLOR. Tento příkaz vybarví téměř libovolný obrazec zvolenou barvou. Používá se rekursivní funkce přizpůsobená procesoru 6502 a úplnost vybarvení obrazce je omezena dostupnou pamětí. V extrémních případech může být zapotřebí až 90 kB paměti RAM. I malé a jednoduché obrazce mohou dočasně zabrat i několik stovek Byte. Když je volná paměť příliš malá, objeví se chybové hlášení ERROR 2 MEM. Příkazy PAINT a TEXT používají vlastní "plot" podprogramy, které umožňují snadné a rychlé spočítání souřadnic sousedních bodů. Nepoužívají rutiny z operačního systému.

TIME \$ viz dále

PAUSE n

Přeruší provádění programu na n padesátin vteřiny. Tento příkaz nahrazuje nepřesné a paměť zabírající časování pomocí prázdných smyček FOR..NEXT. V ATARI Basicu se často pro krátká spoždění používá mocnění ...

A=1^1. Turbo Basic je při mocnění obzvláště rychlý a proto

lze s výhodou použít např. PAUSE 9.

DSOUND kanál,frekvence,zbarvení,hlasitost

Podobá se normálnímu příkazu SOUND s tím rozdílem, že platí pro dva spojené zvukové generátory. Zadání frekvence má potom délku 16 Bite a je možno zadat číslo frekvence v rozsahu 0-65 535 ( místo 0-255).

Výsledná frekvence v HZ se počítá ze vzorce Fvýst =  $1\ 789\ 790/(2 f \text{ frekvence} + 14)$  místo původního Fvýst =  $63\ 921/(2 f \text{ frekvence} + 2)$ . Hodnoty pro počítače vyroběné v USA a v Evropě se mohou lišit - různé frekvence hod. kmitočtů pro odvození televizní frekvence pro normu NTSC a PAL.

SOUND

DSOUND

Zkratka pro FOR I=0 TO 3:SOUND I,0,0:NEXT I

CLOSE - zkratka pro FOR I=0 TO 7:CLOSE#I: NEXT I

F u n k c e :

DPEEK(adr)

PEEK s dvojitým Bytem odpovídá PEEK adr 1 256.PEEK adr+1

INKEY \$

Když se stiskne klávesa, dostane proměnná INKEY\$ odpovídající znak. Když nedojde ke stisku klávesy obdrží prázdný string " ". Tímto způsobe, se nechá zpracovat stisk klávesy, aniž by se přerušil program.

INSTR(A\$, B\$)

INSTR(A\$, B\$, I)

Hledá A\$ v delším řetězci B\$. Když je nalezen, vydá pozici B\$ v A\$, jinak je vydána 0. I nastavuje index(pozici) od níž se začíná hledat.

UINSTR(A\$, B\$)

UINSTR(A\$, B\$, I)

Podobá se příkazu INSTR. Nejsou uvažovány bity 7 a 5 jednotlivých znaků. Např. při hledání "MODEM" se tak nechá nalézt "Modem" nebo "MoDeM" nebo i odpovídající inverzní text. Jako vedlejší efekt jsou při hledání čísel nebo odstavců čísel nalezeny speciální znaky jako ( "!"= CTRL A, "0"= CTRL P atd.)

ERR

zkratka pro PEEK(195) pro určení čísla chyby

ERL

zkratka pro PEEK(186)+256.PEEK(187) nebo DPEEK 186 pro určení čísla řádku na němž došlo k chybě. ERR a ERL mohou být výhodně použity v TRAP podprogramech.

TIME

Speciální proměnná, obsahuje čas od vnitřního časovače RTCLCK a je vydáván obsah paměťových buněk 18 až 20 v padesátinách sec.

TIME\$

speciální proměnná udává čas jako šestimístný řetězec ve tvaru hh mm ss  
hh - hodina 00 až 23, mm - minuta 00 až 59, ss - vteř. 00-59.

TIME\$ =

slouží k nastavení času. TIME 15 15 20 nastaví hodiny na 15 hod 15 min 20 sec. Proměnná TIME se nedá přímo změnit, je nutno použít buďto TIME\$ nebo odpovídající POKE do paměťových míst 18 až 20. Hodiny mají celá zcela přesně, protože frekvence s kterou jsou v ATARI generovány TV obrazy, není přesně 50 Hz a TIME\$ je odvozen od této frekvence.

FRAC(exp)

tato funkce vydá část čísla za desetinou tečkou. Pozor FRAC(exp) není vždy stejně s výrazem exp-INT(exp). Je to z toho důvodu, že INT vydá celé číslo menší než exp-INT(-0.3)=-1 proti tomu FRAC(-0.3)=-0.3

TRUNC(exp)

je to doplňková funkce k FRAC, vydá celočíselnou část čísla. TRUNC(-0.3)=0

RND

v turbo Basicu můžou být vynechány závorky a nula za RND, tím se zkrátí zápis

RND(n)

je to zkratka pro TRUNC(RND(0).n) vydá celé náhodné číslé od 0 do n (nula je zahrnuta, n ne).

### HEX\$(exp)

je to podobné STR\$. Funkce HEX\$ mění celá čísla(exp), kde 0 #exp = 65 535 na hexadecimální řetězec. Když je číslo menší než 256 je vydán dvou jinak čtyřmístný řetězec.

### DEC(sexp)

je to funkce podobná VAL\$. Je to funkce opačná k HEX\$. Řetězec sexp je převeden na decimální číslo. Pokud má řetězec sexp více než 4 platná hex.místa jsou převedena jen poslední čtyři.

### \$ aaaa

Hexadecimální čísla v programu. Příklad:

```
FOR I=$ 0600 TO $ 067F:READ A: POKE I,A:NEXT I  
místo FOR I=1536 TO 1663:READ A:POKE I,A: NEXT I
```

### & - binární AND

### ! - binární OR

### EXOR

### binární EXCLUSIV OR. V

Výše uvedené operace pracují s 16 bitovými celými čísly. Tedy s čísly mezi 0 až 65 535 a ne s booleovskými hodnotami 1 nebo 0 jako běžné operátory AND, OR, NOT.

### DIV

dělení bez zbytku. a DIV b odpovídá TRUNC(a/b)

### MOD

dělení MODULO se zbytkem po dělení a MOD b odpovídá a-b \*TRUNC(a/b)

### %0, %1, %2, %3

\* čísla 0 až 3 jsou definována jako konstanty. Použití čísla také (\$ aaa) zabere v programu vždy 7 Byte, použité proměnné nezávisle na délce jména vždy 1 nebo 2 Byte. Také použití %0 až %3 zabírá 1 Byte, ale nezabírá místo v tabulce proměnných, která má omezenou kapacitu 256 proměnných

Ve stringu nebo mezi uvozovkami je možno v Turbo Basicu použít " což odpovídá CHR\$(34), je ale nutno použít uvozovku dvakrát. Příklad : ?"TEXT" "TEST" vyvolá tisk TEXT"TEST.

Turbo Basic mění při psaní programových řádků znaky malých písmen na velká, inverzní znaky na normální. To neplatí mezi uvodzovkami. Není tedy nutné přepínání mezi malými a velkými písmeny. V názvech proměnných a procedur je možno kromě cifer a malých a velkých písmen použít i znak podtržení   .

DIR

DIR"D1:\*./\*

tisk seznamu (obsahu) diskety na obrazovku. DIR"D1:\*./\* vypíše všechny fajly dískety 1. DIR"D1:A\*./\* vypíše všechny fajly začínající písmenem A. D1:\*./\* může být vynecháno, potom je automaticky nastaveno D:\*./\* podobně jako v DOS menu při A RETURN RETURN. Při tomto příkazu jsou přípustné řetězcové proměnné stejně jako u následujících příkazů.

RENAME"D:OLD,NEW"

přejmenuje fajl OLD na NEW. Odpovídá XIO 32, #7,000, "D2:OLD,NEW nebo volbě DOS menu E

DELETE"D:FILE "

vymaže fajl, odpovídá XIO 33 nebo příkazu DOS menu D

LOCK"D:FILE "

chrání fajl před přepsáním. Odpovídá XIO 35 nebo příkazu DOS F

UNLOCK"D:FILE "

Chrání fajl před přepsáním. Odpovídá DOS příkazu G nebo XIO 36

BLOAD"D:FILE.OBJ"

Uloží binární fajl. Odpovídá DOS příkazu L s následujícím IN za jménem fajlu.

BRUN"D:FILE.OBJ"

Nahraje binární fajl a spustí jej, když fajl obsahuje RUN adresu. Odpovídá DOS menu příkazu L bez závěrečného IN.

Tímto souhrnem jsou objasněny dostatečně funkce a příkazy Turbo Basicu. Samozřejmě jsou k dispozici ještě všechny normální příkazy, které známe z normálního Basicu. S Turbo

Basicem je programování ATARI 800XL a 130XE opravdovou radostí. S Turbo Basicem je u ATARI 800XL k dispozici pro programátora 34 021 Byte, což je o 1 747 Byte více než s normálním Basicem. Turbo Basic je slučitelný s normálním basicem. K obtížím může dojít tehdy, použijete-li v basic programu subrutiny, které se odvolávají na rutiny zabudovaného normálního basicu.

Program Turbo Basicu příklad č.1 :      Příklad č.2:  
10 REM "ELSE" TEST PROGRAM                  10 REM "REPEAT..UNTIL"  
20 X=1    20 X=9  
30 IF X <> 5                                30 REPEAT  
40 GOTO 80                                    40 PRINT X  
50 ELSE                                        50 X=X-1  
60 GOTO 100                                    60 UNTIL X=-1  
70 ENDIF                                        70 PRINT "END"  
80 X=X+1                                        80 END  
90 PRINT X:GOTO 30  
100 PRINT "END"  
110 END

Příklad č. 3:

```
10 REM "WHILE..WEND" TEST PROGRAM
20 DIM LS(5)
30 WHILE LS <> "KONEC"
40 READ X,LS
50 PRINT X,LS
60 WEND
70 PRINT "END"
80 END
100 DATA 3,TRI
110 DATA 2,DVA
120 DATA 1,JEDNA
130 DATA 0,KONEC
```

Příklad č. 4:

```
10 REM "DO..LOOP" DEMO
20 X=1
30 DO
40 PRINT "ATARI COMPUTER"
50 IF X=10 THEN EXIT
60 X=X+1
70 LOOP
80 PRINT "END"
90 END
```

Pro správné porozumění demonstračních programů zadej před RUN : TRACE +. Tento příkaz bude trasovat program tak, že bude vypisovat čísla řádků, které počítač prochází. Pro zrušení tohoto modu: TRACE -.

TURBO DIGI :

```
10 REM DIG.TIME
20 DIM A$(6)
30 INPUT "TIME(hhmmss)";A$
40 TIME$ = A$
50 GRAPHICS=19:COLOR 2
60 FOR A=0 TO 1 STEP 0
70 A$=TIME$
80 TEXT 3,3,A$(1,4): TEXT
11,11,A$(5)
90 NEXT A
100 END
```

Přeložil a upravil Ing.Hrdlička Jiří

Uživatelský popis programu A.C.E. - The ATARI Casette  
Eunanter:-----

Když je ACE program zaveden do počítače, chrání sám sebe, takže můžete vložit normální BASIC program bez ovlivnění (poškození) ACE. Za normálních podmínek zůstává v paměti připraven k použití, dokud nevypnete ATARI.

!!! Nelze použít současně DOS a ACE.

Zavedení do počítače:

Nahrajete pomocí tlačítka START při zapínání počítače. Asi po 50 vteřinách nahrávání se objeví 3 nabízené volby. Které příkazy ACE jsou v které volbě obsaženy, je popsáno dále. Některou ze tří voleb zvolíte stisknutím A,B nebo C. ACE potom zavede zbývající část pásky.

Volba "A" - zabírá 2298 Bytů paměti, nahrává se 61 secund.

Dovoluje pojmenovat programy uschované na kazetě, ukládat, natáhnout, vertifikovat a spouštět pojmenované podprogramy(soubory), prohlížet kazetu na které jsou pojmenované programy, nechat blikat kurzor, nebo kterékoliv inverzní znaky.

Volba "B" - zabírá 3496 Bytů, nahrává se 84 sec. Obsahuje všechny možnosti jako volba A, navíc poskytuje velmi rychlé přečíslování čísel řádků programu, automatické číslování řádků, výpis proměnných a jejich hodnot.

Volba "C" - zabírá 4148 Bytů, nahrává se 96 sec. Poskytuje všechny vlastnosti B volby, navíc ukládá na pásku, natahuje z pásky a vertifikuje binární soubory, pro-

vádí konverzi dekadického-hexadecimálního-binárního čísla. Vyčísluje paměťové ukazatele (tj. vyčísluje hodnotu 2 Bytů na zavedené adrese).

Využívání ACE - všeobecné informace:

ACE rozlišuje vaše příkazy dle jednoduchého označení: znakem "@". Proto, kdykoliv chcete požádat ACE, aby něco udělal, musíte váš příkaz začít znakem @. Navíc znak musí být ve sloupci nejvíc vlevo, tj na začátku řádky.

@ znak byl vybrán proto, že se jinak v 1. sloupci nikdy nevyskytuje. Pokud jej chcete zaměnit za jiný znak, použijte POKE 2047, ATASCII kód nového znaku. Např.: Chcete-li místo @ používat "!", zadejte POKE 2047, 33

Volba "A" + vlastnosti a příkazy:

@CSAVE "Jméno" - uschovává program z paměti na kazetu současně se zvoleným jménem (až 16 znaků dlouhým). Takovýto program může být nahrán zpět do paměti i normálním příkazem CLOAD.

@CLOAD "Jméno"

vyhledává program uvedeného jména na kazetě a nahraje jej do paměti.

@CVERIFY "Jméno"

vyhledává program uvedeného jména a verifikuje (tzn. ověří shodu programu na páisce a v paměti).

@CRUN "Jméno"

vyhledává program uvedeného jména. Natáhne do paměti a spustí.

@CATALOG

prohledá celou kazetu, všechny pojmenované programy, které jsou nalezeny jsou vypsány na obrazovku.

@VERIFY

verifikuje program, který byl předtím uschován LIST "C:"

@FLASH

způsobí blikání kurzoru. Rychlosť blikání lze řídit příkazem POKE 2421,X - kde X je číslo mezi 1-255. Význam má nas-

tavení jediného Bytu, tj.: X=1, 2, . , 8, 16, 32, 64 a 128.

**@FLASH X**

X=0 bliká pouze kurzor

X=1 inverzní znaky se objeví v norm. zobraz. a blikají

X=2 inverzní znaky blikají mezi normálním a inverzním zo-  
brazením

X=3 inverzní znaky se zobrazí normálně a blikají mezi nor-  
málním zobrazením a plným bílým polem

X=4 všechny znaky blikají mezi normálním zobrazením a zo-  
brazením "vzhůru nohama"

X=64 inverzní znaky blikají mezi inverzním zobrazením a  
plným bílým polem.

**@OFF - zruší všechny FLASH příkazy**

V o l b a B - vlastnosti a příkazy:

zůstávají v platnosti  
všechny příkazy popsané ve volbě A, navíc lze použít:

**@DUMP**

tiskne na obrazovce seznam všech proměnných použitých v  
programu, který je právě v paměti počítače a to společně  
s hodnotami těchto proměnných. Pokud byl program dříve  
uschován na pásmu se jménem, je v seznamu i toto jméno.

**@AUTO**

aktivuje automatické číslování řádků. Pokud není v paměti  
žádný program, začíná číslovat od 100. Je-li v paměti pro-  
gram, začíná číslovat od posledního řádku plus specifikova-  
ný přírustek. Není-li přírustek specifikován, jeho implicit-  
ní hodnota je 10.

**@AUTO s,i**

jako AUTO, ale počáteční číslo řádku a jeho přírustek i  
je definován.

**@AUTO s**

dtto, ale implicitní hodnota přírustku je 10

Jestliže je aktivována funkce AUTO a narazí na číslo řád-  
ku, který již existuje, AUTO se samo vypne.

### @REPORT

pokud se provádí přečíslování čísel řádků programu, nelze přečíslovat např. GOTO.LINE, kde LINE je proměnná. Příkaz REPORT vypíše všechny takovéto příkazy, navíc upozorní na odkazy neexistující číslá řádků.

### @RENUMBER s,i

přečíslování čísel řádků v programu v paměti a to tak, že 1. řádek programu bude mít číslo s a přírustek bude i. Přečísluje i příslušné odkazy na čísla v řádku v programu (GOTO,GOSUB, ON...GOTO,ON...GOSUB,LIST,RESTORE,TRAP a THEN). Odkazy na neexistující řádky jsou přečíslovány na 32 767 pro snadnou kontrolu.(implicitně s=100,i=10),RENUMBER s (implicitně i= 10), RENumber i (implicitně s=100).

Ke všem tvarům lze přidat " ,E" potom jsou všechny případy popsané v případu REPORT označeny.

### POZNÁMKA:

Je-li užit příkaz TRAP 40 000, který slouží pro zrušení příkazu TRAP, bude číslo řádku změněno na 36 767, což je číslo, které ještě stále zruší TRAP.

### V O L B A C - vlastnosti a příkazy:

všechny vlastnosti A+B

jsou zachovány, navíc :

### @CSAVE x,y

uschová objekt, soubor z paměti od adresy x do y. (mohou být i v hexadecimálním tvaru začínající znakem \$ např.: \$A000).

### @BLOAD

natáhne objekt, soubor z kazety do paměti. Dolní a horní adresa tj. x a y z příkazu CSAVE jsou zobrazeny.

**@BVERIFY** - ověří binární soubor uschovaný na kazetě oproti paměti.

### @VAL x

zobrazí současně hodnotu x ve tvaru hexadecimálním, dekadickém a binárním.

@POINTER x

zobrazí hodnotu ukazatele (dvou bytů za sebou v paměti)

- rk -

Prostorová grafika s rotací .

Použití:

Program umožňuje kreslit na obrazovku tělesa v tzv. Mongeově promítání ( v osách x a y, souřadnice z vyjádřeny číselně).

Postup:

Ve středu obrazovky se así po 30 sec. vysvítí bod a v textovém okénku svítí souřadnice počátku ( $x=0, y=0, z=0$ ). Tlačítky  $\rightarrow, \leftarrow, \uparrow, \downarrow$  bez CTRL se zvolí směr pohybu bodu a po dobu stisknutí tlačítka START se svítící bod pohybuje zvoleným směrem, současně se mění příslušná souřadnice v textovém okénku. Po dosažení požadované polohy bodu (vrcholu tělesa) se zapíší souřadnice tlačítkem "." (bod). Zvolí se opět směr a pokračuje do dalšího bodu. Je-li kdykoliv potřeba změnit souřadnice z, stiskne se tlačítko z a číselně se zadá velikost souřadnice. Po dosažení dalšího vrcholu tělesa se příslušná "hrana" zaznamená tlačítkem ","(čárka). Na displeji se provede spojení s předchozím bodem, nebo s posledním bodem minulé spojnice. Je-li třeba odstranit chybnou čáru, stiskne se " $\emptyset$ " a opakuje "kreslení" chybné čáry. Návrat do režimu normálního kreslení je " $\uparrow$ " tlačítkem. Je-li potřeba kreslit kruhový oblouk, zafixuje se "." počáteční bod, nebo je brán koncový bod poslední čáry. Svítícím bodem se najede na budoucí střed kružnice a tlačítkem "k" se přejde do režimu kružnice. Je nutno zadat směr  $\rightarrow$  doprava, nebo  $\leftarrow$  do leva a tlačítkem START je postupně po  $10^{\circ}$  kreslen oblouk. V režimu kružnic je rovněž možno umazávat " $\emptyset$ " i kreslit "l". Výstup z režimu kružnic je tlačítkem "K" do režimu přímek.

POZOR! U tělesa (např.krychle) je nutno zobrazit i hrany rovnoběžné s osou z, tedy např. při  $z=0$  zaznamenat jeden vrchol, změnit po stisku z z-ovou souřadnici a stiskem "," zaznamenat hranu. Na displeji se nic nového nezobrazí, ale

je zaznamenána v paměti počítače a zobrazována při rotaci tělesa kolem osy x nebo y. Po ukončení kreslení tělesa je možno přejít tlačítkem "R" do režimu rotace. Stiskem x,y nebo z se uvede do chodu postupné otáčení tělesa po 15°. Zastavit rotaci lze libovolným tlačítkem, změnit směr lze tlačítkem "-" (minus) a osu rotace změnit x,y nebo z.

```
1Ø GR.8:SE.2,1Ø,2:DIM A(499,5):DØG
2Ø X=16Ø:Y=95:I=Ø:BA=1:SIG=SIN(15):COG=COS(15):z=Ø:C0=1
3Ø F.J=Ø TO 499:F.K=Ø TO 5:A(J,K)=Ø:N.K:N.J:GOS.17Ø
4Ø GOS.18Ø:IF PEEK(764)=5 THEN POKE 764,255:G.23Ø
5Ø IF PEEK(764)=14 THEN DX=Ø:DY=-1
6Ø IF PEEK(764)=15 THEN DX=Ø:DY=1
7Ø IF PEEK(764)=6 THEN DX=-1:DY=Ø
8Ø IF PEEK(764)=7 THEN DX=1:DY=Ø
9Ø IF PEEK(764)=4Ø THEN 35Ø
1ØØ IF PEEK(764)=32 THEN GOS.32Ø
11Ø IF PEEK(53279) <> 6 THEN 4Ø
12Ø C.BA:PL.X,Y:X=X+DX:Y=Y+DY:IF X<Ø THEN X=Ø
13Ø IF Y<Ø THEN Y=Ø
14Ø IF X>319 THEN X=319
15Ø IF Y>191 THEN Y=191
16Ø LOC.X,Y,BA:GOS.17Ø:G.11Ø
17Ø C.1:PL.X,Y:POKE 656,2: POKE 657,1:?"X= ⌂"; INT(X)-16Ø,
    "Y= ⌂"; INT(Y)-95, "Z=⌂"; Z:RET.
18Ø IF PEEK(764)=31 THEN C0=1
19Ø IF PEEK(764)=5Ø THEN C0=Ø
2ØØ IF PEEK(764)=23 THEN POKE 764,255:POKE 656,2:POKE 657,23:
    ?"⌂⌂⌂";:POKE 657,23:I.Z
21Ø IF PEEK(764)=34 THEN POKE 764,255:XB=X:YB=Y:A(I,Ø)=X:
    A(I,1)=Y:A(I,2)=Z:BA=1
22Ø RET.?
23Ø R=SQR((X-XB)^2+(Y-YB)^2):XS=X:YS=Y: IF XB=XS AND
    YB>YS THEN A1=9Ø:G.26Ø
24Ø IF XB=XS AND YB<YS THEN A1=27Ø:G.26Ø
25Ø A1=ATN((YB-YS)/(XB-XS)): IF XB<XS THEN A1=A1+18Ø
26Ø GOS.18Ø: IF PEEK(764)=7 THEN AL=1Ø
```

```
270 IF PEEK(764)=6 THEN AL=-10
280 IF PEEK(764)=5 THEN POKE 764,255:G.40
290 IF PEEK(53279)<>6 THEN 260
300 A1=A1+AL:X=INT(XS+R*COS(A1)+.5):Y=INT(YS+R*SIN(A1)+.5)
310 GOS.320:G.290
320 POKE 764,255:C.CO:PL.XB,YB:DR.X,Y:XB=X:YB=Y: A(I,4)=Y:
    A(I,5)=Z:IF CO=& THEN A(I,3)=-X:G.340
330 A(I,3)=X
340 I=I+1:BA=1:A(I,&)=X:A(I,1)=Y:A(I,2)=Z:GOS.170:RET.
350 IF PEEK(764)=14 THEN SIG=-SIG: POKE 764,255
360 IF PEEK(764)=22 THEN DY=95:DZ=&:IY=1:IZ=2:G.400
370 IF PEEK(764)=43 THEN DY=&:DZ=16&:IY=2:IZ=&:G.400
380 IF PEEK(764)=23 THEN DY=16&: DZ=95:IZ = 1: G.400
390 G.350
400 F.J=& TO I-1:F.K0 TO 3 STEP 3: SG=SGN(A(J,3)):A(J,3)=
    ABS(A(J,3))
410 Y=A(J,IY+K)-DY:Z=A(J,IZ+K)-DZ:A(J,IY+K)=Y*COG-Z*SIG+DY:A(J,IZ+K)
    A(J,IZ+K)=Y*SIG+Z*COG+DZ
420 A(j,3)=SG * A(J,3):N.K:N.J:GR.8: SE.2,10,2
430 F.J=& TO I-1:IF A(J,3)>=& THEN C.1:G.450
440 C.&
450 PL.A(J,&), A(J,1): DR.ABS(A(J,3)), A(J,4):N.J:G 350
```

Připravil Ing. Petr Valka

---

#### Popis hry S O L O F L I G H T - 1983 by Sid Meier

Solo flight je simulace v reálném čase, která umožní získat zkušenosti s pilotováním sportovního letadla. Musíš vzlétnout, přistát, letět podle přístrojů, řešit nebezpečné situace, vzdorovat tříbulencím a špatnému počasí. Zahraj si také Mail Pilot(poštovní pilot), kde závodíš s časem, mechanickými poruchami a špatným počasím.

Program se zavádí přes START + OPTION.

Volba módu:

Tlačítky OPTION a SELECT určíš mód a úroveň obtížnosti:

OPTION : FLYING (létání v oblasti)

dalším zmáčknutím OPTION volíš jednou oblast

a) Kansas - rovina

b) Washington - vrchovina

c) Colorado - hornatý terén

SELECT v tomto módu volí obtížnosti: CLEAR(jasno), LANDING(rutina pro přistávání), WINDY(větrno) a IFR(let dle přístrojů)

MAIL RUN (poštovní trasa - další volbou OPTION)  
stejně jako v módu FLYING lze i v tomto volit jednou oblast (Kansas, Washington a Colorado).

SELECT i zde volí stupně obtížnosti. Nejsnadnější je STUDENT, dále PRIVATE, SENIOR a nejobtížnejší COMMAND.

!! Abys mohl hrát stiskni START dříve než měřič času dosáhne 0.

P O Š T O V N Ě T R A S A :

Tato hra testuje Tvé letecké schopnosti a rozhodnost. Cílem je doručit 5 balíků pošty v co nejkratším čase. Když si vybereš MAIL RUN objeví se mapa (jen u diskové verze). Stiskni START a pokračuj. OPTION používáš pro nalezení pošty. Můžeš naložit kolik pošty chceš, ale každý balík je zátěž a ztěžuje let. Doporučujeme 1-2 balíky. Můžeš natankovat palivo, (SELECT). Palivo je také zátěž, tank má obsah 64 galónu.

Když jsi připraven na cestu, stiskni START. Znovu se objeví mapa (jen u diskety). Dobře si prohlédni charakter krajiny, navigační věže VOR, blízká letiště. Až budeš připraven ke startu opět stiskni START. Stisknutím OPTION se můžeš vrátit pro další poštu a palivo, když přiletíš na určené letiště a přistaneš. Tvé přistání je vyhodnoceno: Pomalé přistání(SPEED) a opatrné přistání(RATE). Přičtou se Ti body i za úroveň obtížnosti za vybranou oblast - Kansas=500, Washington=1000 a Colorado=1500, za spotřebovaný čas (0-1000). Jestli se zřítíš, musí se zjistit.

tit proč. Zřícení letadla při přistávání(CRASH LANDING) znamená, že jsi udeřil příliš tvrdě o zem nebo nevysunul přistávací podvozek. Špatná přistávací dráha (MISSING RUNWAY) znamená, že jsi nepřistál na přistávací dráze. GROUND LOOP znamená, že jsi chtěl otočit letadlo příliš ostře. STALLED znamená, že jsi letěl moc pomalu a ztratil jsi rychlosť. U diskové verze se ukáže mapa a trasa. Zmáčkní start a pokračuješ v Poštovní trase. Hra končí, když všech 5 balíků je doručeno, nebo když se letadlo zřítí.

Během hry se počasí zhoršuje. Zesílí vítr, mohou přijít mraky a může se objevit turbulence. Na vyšších úrovních obtížnosti je Tvé letadlo náchylné k mechanickému poškození. Motor se může přehřát(informuje o tom TEMP). Různé přístroje se mohou porouchat. Všechny poruchy se opraví, když přistaneš na letišti.

#### Přístroje:

Dolní polovina obrazovky obsahuje jednotlivé letecké přístroje.

Vlevo je výškoměr. Každá značka značí 1000 stop pro malou ručičku a koresponduje s celkovými změnami velké ručičky.

Vpravo je RYCHLOMĚR, který ukazuje od 0 do 1,80 uzlů. Malý kruh uprostřed je UMĚLÝ HORIZONT, který indikuje polohu letadla vzhledem k horizontu.

Vertikální pruh uprostřed je regulátor plynu. Maximální síla je nahore, nulova dole.

Čtyři digitální indikátory vlevo dole jsou velice důležité:

PITCH - naklonění letadla. Kladné hodnoty znamenají, že špice směřuje nahoru, záporná hodnota, že špice směřuje dolů. Přesná kontrola je nutná pro zajištění míry stoupání a klesání.

PLAPS - stupeň nastavení přistávacích nebo startovacích klapek.

HEAD - směr na kompasu. 0° znamená sever (N), 90° východ(E), 180° jih(S), 270° západ(W).

Např.: 330 NE = 330° severovýchodně

CLIMB - vertikální rychlosť stoupáni. Kladné hodnoty znamenají, že získáváš výšku, záporné, že ztrácíš. (Např.: -2 = strmé klesání)

Měřič paliva je vpravo dole.

Indikátor svítící vlevo uprostřed upozorňuje na přehřátý motor ( TEMP ).

Dvě světla vparvo uprostřed upozorňují, že podvozek je vytažen (L) a že brzdíte (B).

Navigační přístroje jsou vpravo dole. Dva indikátory (VOR) udávají přesnou polohu vzhledem k radiomajákům.

VOR 1 - azimut od radiomajáku 1 (270 = na západ od 1)

VOR 2 - azimut od radiomajáku 2 (45 = na SV od 2 )

Pomocné ukazatele:

WIND - vítr = udán směr(úhel) a síla(rychlosť) 270 2

CEILING - spodní hranice mraků (např.: 1000 = jasno)

VISIBILITY - viditelnost (dohlednosť) max. - 5

Ovládání :

Kromě ovládání joystickem, řada příkazů může vstupovat přes klávesnici.

Rgeulátor paliva: Číslo 0-9 (0-žádná síla, 9-plný tah)

Podvozek : stisknutím "L" zvedneš nebo spustíš podvozek.

Brzdy : stisknutím "B" zapneš nebo vypneš brzdění.

Start. a přistávací klapky: "F" řídí nastavení klapiek.

· postupně 0°-20°-40°-0°

Pohled : Použij tlačítka ovládání kurzoru. Šipka vlevo vpravo umožní pohled z pravého a levého okénka. Šipka dolů je pro pohled do zadu. Šipkou vpřed se vrátíš do předu. Nezapoměň stisknout současně CTRlnebo SHIFT, když použiješ tyto klávesy.

Přestávka: Stisknutím "P" přerušíš hru. Pokračuješ stiskem libovolné klávesy.

Nebezpečí: "E" jestliže chceš způsobit poruchu zařízení.

Sklouznáti po křídle: Nakloň letadlo, stiskni joystick a sveď letadlo do klouzavého letu zatočením směrového kormidla a křidélka na nosných plochách letadla. Letadlo klouže proti větru a klesá.

### Basic Flight - základy letu:

Během letu užíváš dvojí řízení. Joystikem měníš polohu letadla a množstvím paliva rychlosť, sílu motoru. Při startu a šplhání je třeba největší síla, malá při přistávání. Mnoho začátečníků používá náklonu k řízení rychlosti a množství paliva k řízení výšky. Tryskoví piloti užívají náklon letadla k přistání. Používají množství paliva pro řízení rychlosti. Existují vztahy mezi náklonem letadla, množstvím paliva, rychlosti a úhlem letu. Nemůže být provedena změna jednoho faktoru bez ovlivnění ostatních. Když zvedneme nos letadla nahoru, rychlosť se sníží a naopak. Také redukci paliva letadlo nakládání se dolů. Pro bezchybný let musí být brán ohled na všechny tyto faktory.

### Přístroje pro zaměřování:(Instrument aparatur)

Přístroji může být zaměřena vysoká výška, nízká výška, přesnost. Zaměření vysoké výšky sestává z přístroje, který udržuje pevnou výšku (IAF) a určené letecké trasy, která může být narušena fyzickými překážkami (hory v okolí Donveru) nebo hodně frekventovanými leteckými linkami (Kennedyho letiště) v New Yorku. Letadlo to překoná snížením paliva, klesnutím na konstantní výšku a návratem do daného směru navedením na FAF. To je zaměřovač na přistávací dráhu ve vzdálenosti 5-10 mil od letiště. Cesta z FAF vede obvykle přímo na letiště. Pilot klesne na minimální výšku (MDA) 200-500 stop 30-45 sec. před MAP. MAP může být definován počítáním času a VOR(řídící věže) na dosažení konce přistávací dráhy konstantní rychlosti. Jestli je příslušná dráha viditelná a pohodlná k přistání, pilot může přistát. Jestli tyto podmínky nejsou splněny, pilot musí přidat sílu, přeletět letiště a zkusit to znova, nebo letět na jiné letiště.

Zaměření malé výšky: je obdobné jako u vysoké výšky. Rozdíl je v tom, že zaměření nízké výšky je obvykle prováděno blíže letiště a při nižší rychlosti. Obojí má za úkol navést letadlo bezpečně na přistání, před dosažením MAP.

### Zaměření přesnosti

Zaměření vysoké výšky, nízké výšky a přistávání po pravouhelníku je definováno jako nepřesné zaměření, protože pilot nedostává přesné informace pro let. Přesného zaměření lze dosáhnout kombinací zaměřením, kombinací konečné stanice VOR a MAP a informací ILS (přistávací systém). Indikátor ILS je umístěn vpravo na řídícím panelu. Horizontální čára na řídícím panelu ukazuje správnou přibližovací výšku. Symbolická pozice letadla znázorňuje výšku letadla ve vztahu ke klouzavému letu. Klouzání je normálně pokles o 3-5 stupňů z FAF na DH (Decision Height). DH je stejně jako MDA, ale je definovana jako minimální výška dosažitelná klouzavým letem. Jestliže symbol letadla je nad horizontální čárou, letadlo je nad klouzáním. Trochu redukuj sílu, skloň letadlo a klesej 200-300 stop za minutu. Jestli je letadlo uprostřed, (klouzavého pohybu), přidej sílu (palivo), uprav míru klesání na 400-500 stop za minutu. Jestli letadlo je pod horizontální čarou, letadlo je pod mezí klouzavého letu. Musíš redukovat klesání a vyšplhat na výšku stanovenou pro klouzavý let. Při přistání hledej na DH a runway musíš mít na dohled. Jestli dosáhneš MDA nebo DH a runway není v dohledu, proved nové přiblížení nebo let na jiné letiště. ILS je automatizovaný přistávací systém, který navádí pilota ve špatném počasí. Moderní letadla mohou přistát i za nulové viditelnosti a při nulové výšce oblaků.

U Solo flight je minimální výška mraků 250 stop a viditelnost 1/4 mile.

Pamatuj! Dobře naváděný pilot přistane na první pokus.

Při několika nepříznivých pokusech mu stoupne hladina adrenalinu, ubývá mu palivo a nejbližší další letiště může být 250 mil daleko. To nejlepší co může pilot udělat, je přistát napoprvé.

Pro návrh vlastního přistání musíš znát:

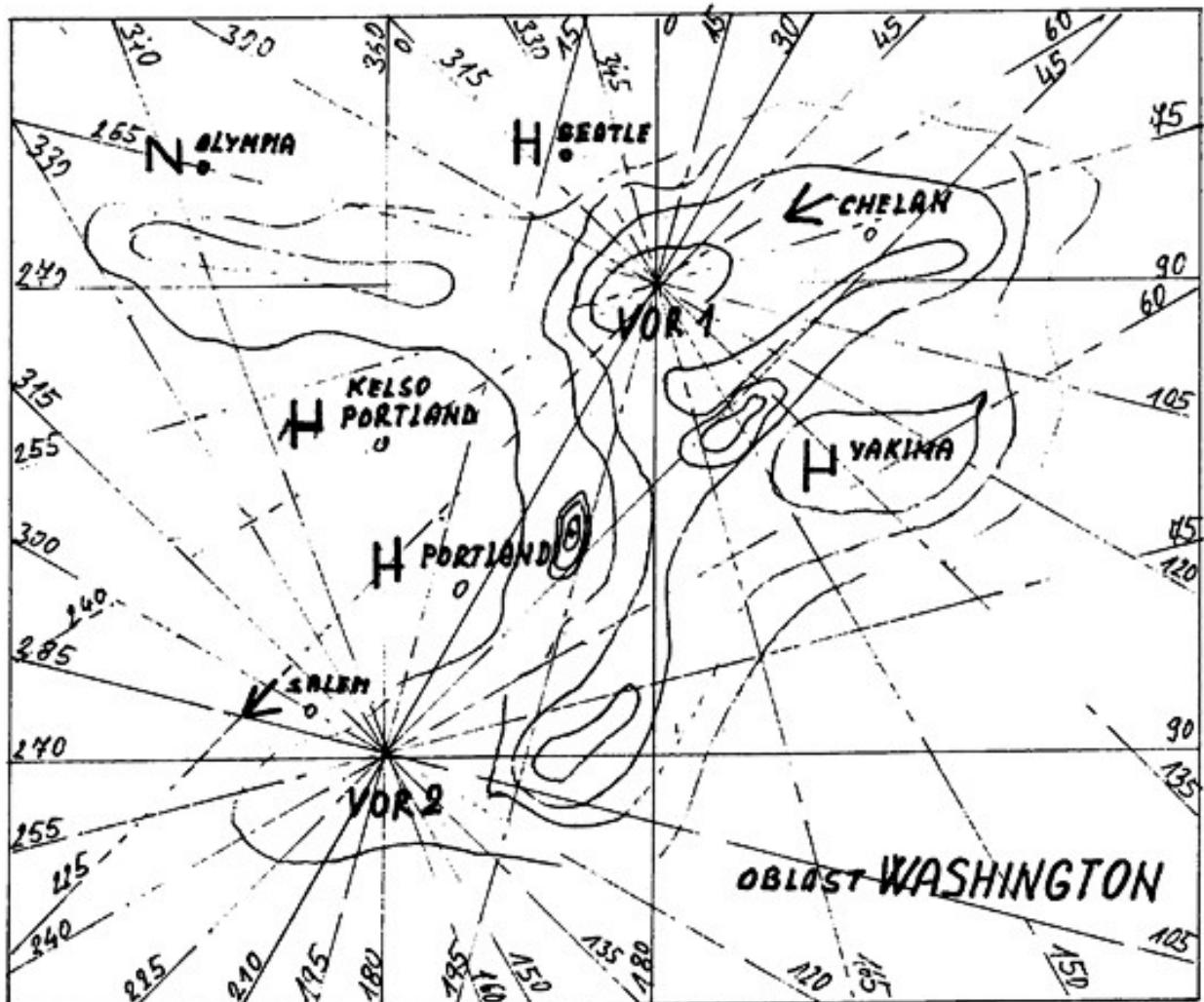
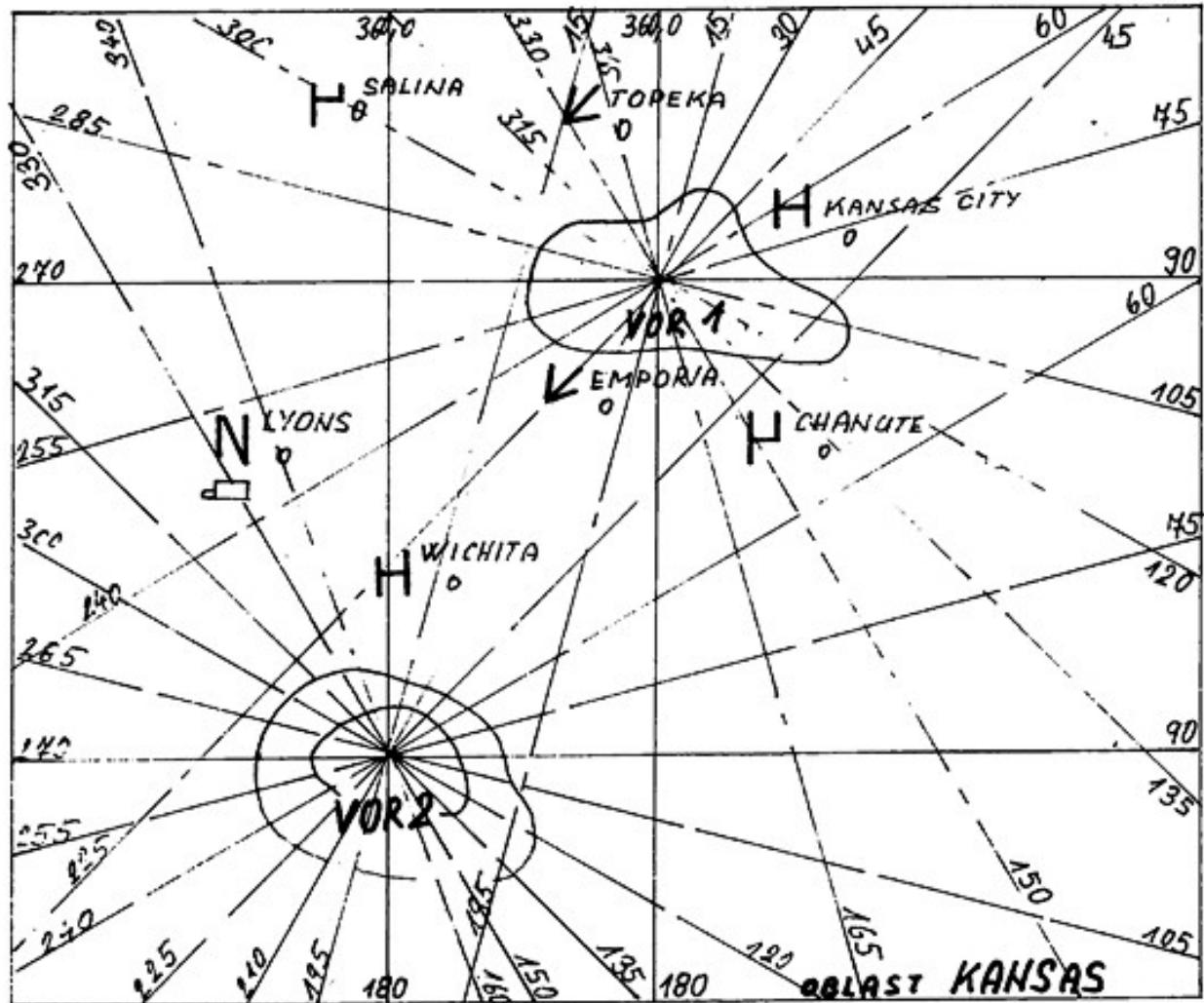
- 1/ konečné nazádení z I na VOR stanici
- 2/ MAP nebo DH definované křížením radiálu, nebo měřením času z VOR.

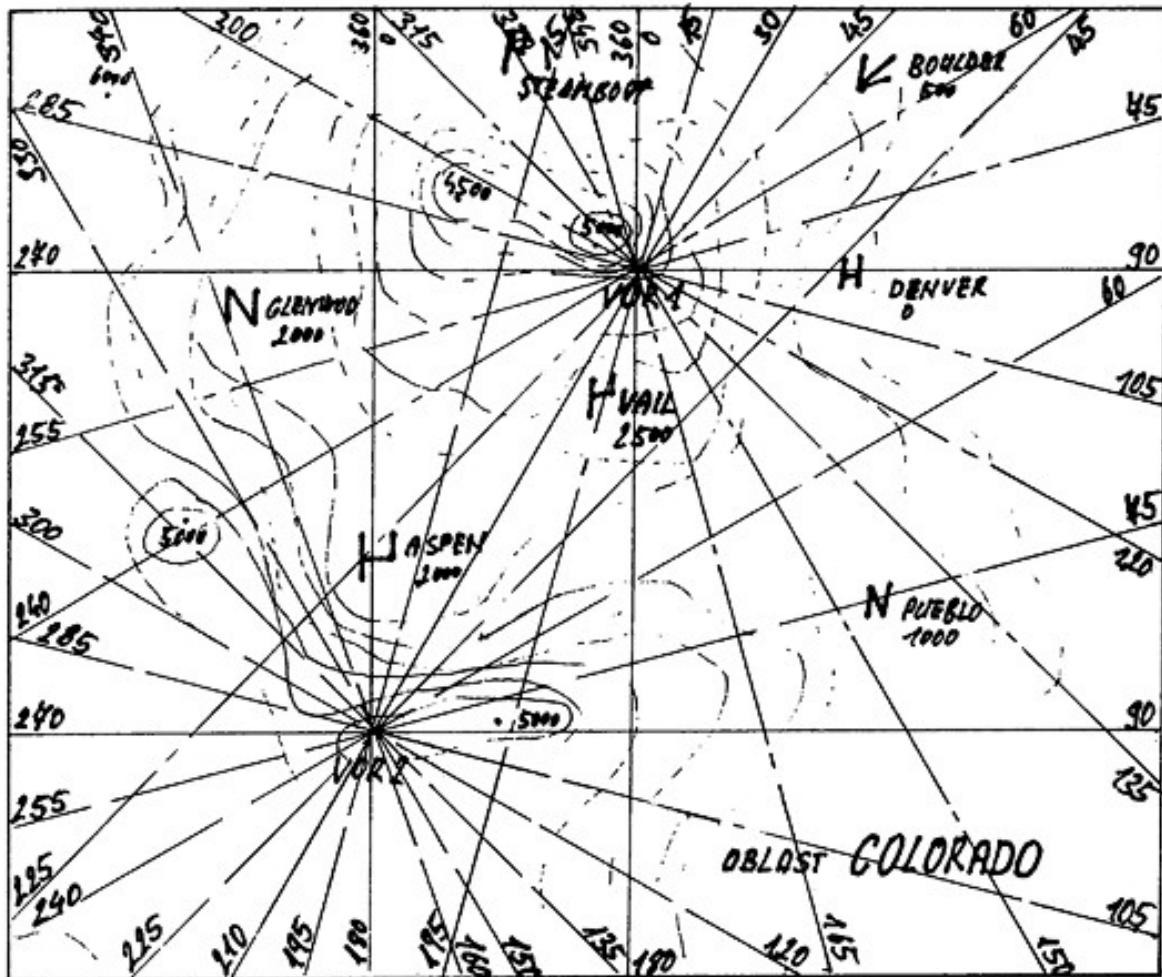
- 3/ FAF definované pomocí VOR
- 4/ IAF (obvykle vyšší než FAF)
- 5/ Cesta (klesání z IAF na FAF)
- 6/ výška vyhrazená pro přistání

#### RADY NA ZÁVĚR:

Najed na konec runwaye (palivo 2-3). Otoč se, vysuň klapky (normálně jsou 20°). Zapni max. tah (9), když dosáhneš 85 uzlů, náklon musí být 8-9 stupňů. Jestli jste už bezpečně ve vzduchu s kladným VVI, zatáhni podvozek. Podvozek se zatahuje asi ve výšce 500 stop. Také zasuň klapky (všimni si, že stoupání se zmenšuje, když klapky jsou vysunuty. Dálší náklon je třeba pro šplhání). Když dosáhneš požadované výšky, redukuj palivo (silu) a vyrovnej letadlo.

Přistávání letadla je nejobtížnější. Jakmile se přiblížíte k určenému letišti, zavedět letadlo nad požadovanou runway. Přiblíž se k letišti ve výšce 1000-1500 stop. Když je uvedeno do činnosti ILS, vystup nebo sestup do prostřed obrazovky. Vysuň podvozek, sniž palivo na 4 a klesej 400-500 stop za minutu. Když se přiblížíš k přistávací dráze, zpomal klesání. Řešení rychlosti je velmi obtížné. Při velké rychlosti přeletíš runway. Při nízké rychlosti se zřítíš. S klapkami natočenými na 20° musíš přistát rychlosti asi 70 uzlů. Bez klapek zvyš konečnou rychlosť o 10 uzlů. Klouzáním ve větru musíš dosáhnout požadované výšky. Jestli klesáš moc pomalu, přidej palivo a dosáhni nulového klesání přesně, když se kola dotknou země. Buď opatrny! Redukce paliva může mít za následek velkou ztrátu rychlosti a zřícení. Jakmile je letadlo na přist. dráze, vypni palivo (stiskni na klávesnici 0) a zapni brzdy (stiskni B) Musíš zastavit letadlo před koncem dráhy. Na krátkých dráhách je to velmi obtížné





#### GREAT AMERICAN CROSS - COUNTRY ROAD RACE

Ihned po nahrání programu do počítače přes "START+ OPTION" se na obrazovce objeví seznam deseti nejlepších závodníků i s jejich časy. Můžete si zvolit soupeře a tedy i čas, který hodláte pokořit, nebo můžete volně soupeřit s počítačem. Poté vám bude dána možnost vybrat si ze čtyř možných tras napříč kontinentem.

- z Los Angeles do Miami
- ze Seattle do Miami
- ze San Francisca do Washingtonu
- cestování po USA.

Směr cesty je vždy od západu na východ. V módu cestování po USA musíte projet všemi městy na mapě. Jakmile si vy-

berete trasu objeví se na obrazovce mapa USA. Vybere-  
te si následující cílové město, zkontrolujete si ces-  
tu, počasí a nastavíte čas odjezdu.

### 1. Příprava na cestu :

#### 1.1. Výběr soupeře

Ze všeho nejdřív se Vás počítač zeptá "Load opposing field from tape?" což znamená .. hodláte si nahrát své soupeře z dalšího pásku?

Jestliže chcete soupeřit s hráči zobrazenými v úvodním seznamu, neboli nechcete-li nahrát pole soupeřů z další pásky stiskněte N/NO. Stisknete-li Y/YES V případě nahrání externího pole soupeřů se po Y objeví na obrazovce "Insert TOP TEN" tape and press PLAY". Během nahrávání bude obrazovka prázdná. Po nahrání programu se zobrazí příslušné pole soupeřů.

#### 1.2. Výběr trasy

V horní části obrazovky je zobrazen název trasy ( 4 trasy viz úvod). Jestliže souhlasíte Y. Po START se objeví mapa.

#### 1.3. Zobrazení mapy (Mapa-mod)

- určení jednotl. cílových měst. Blikající kroužek ukazuje místo, kde se momentálně nacházíme. Blikající šipka ukazuje města, která si lze vybrat jako další cílová města na trase(změna joystikem)
- hodiny v dolním levém rohu ukazují čas odjezdu(startu) Plánujte jej pečlivě tak,aby jste se vyhnuli příjezdům do města v dopr. špičce. Nastavování pomocí OPTION

### 2. S T A R T !

přístroje:na palubní desce

- ukazatel množství PHM(start vždy s plnou)
- timer uprostřed obrazovky odpočítává čas, který máte vymezen na etapu.Ušetřený čas máte k dobru do dalších cest. Nestihnete-li časový limit jste diskvalifikován.
- odometer pod trimrem odpočítává vzdálenost (míle) od cílového města.
- radar nás blikáním upozorní na policejní hlídka na silnici
- RPM (tachometr) informuje o rychlosti jízdy(míle/hod)

Jsme na cestě !!!

-řízení Joystikem: dozadu-brzdění, vlevo vpravo řídí směr jízdy vpravo vlevo. Triger-přidávám plyn.  
-řazení rychlostních stupňů. Čím rychleji jedete, tím se kola rychleji točí, tedy zvyšují se otáčky motoru. Většina motoristů řadí podle ucha-tedy dle stupně revu motoru, někteří dokonce, dokonce podle údaje otáčkoměru. Ten je v našem stroji označen RPM(ot/min). Jestliže otáčky dosáhnou hodnoty 9, pak je třeba přeřadit na vyšší převodový stupeň. Jinak dojde k přehřátí motoru, což Vám obrazovka sdělí nápisem PUSH TU PUMP. Musíte vůz tlačit k nejbližší benz.stanici. Opakováný stisk Trigru. Vlastní řazení provedem po uvolnění Trigru zatlačíme Joystikem krátce dozadu. Ihned uvidíme na otáčkoměru, ručička klesne na minimum.

Čerpací stanice pohonných hmot je každých 100 mil. Díváme-li se dobře na cestu, všimneme si, že seblížící pumpa je inzerovaná nápisem na obrazovce. Chceme-li natakovat stačí zastavit vedle pumpy a sledujete ukazatele stavu nádrže. Jestliže Vám dojde benzín na trati, nebo přehřejete motor musíte dotlačit auto k benzínové pumpě.

Radar v pravém rohu obrazovky Vás včas upozorní, že brzy dojedete a budete muset předjet policejní silniční hlídku. Činěte tak disciplinovaně (rychlosť 64 mil/hod) neboť i v této hře Vás může orgán nepříjemně zdržovat! Jestliže nezvládnete řízení a vyjedete z vozovky nebo štěrjchnete předjížděné vozidlo atd. zkrátka při kolizi budete nemilosrdně zabrzděni a uvrženi do prvního rychlostního stupně. Když smyky a brzdění skončí, přidejte plyn a vrátte se normálně do závodu.

### 3. Dosažení cílového etapového města

"WELCOME" ... Tento nápis Vás přivítá v každém etapovém cíli. Posléze se dostanete do Mapa-modu. Tam bude zobrazen Váš průběžný čas (E.T) a průměrná rychlosť. Blikající kruh ukazuje vaší současnou pozici. Předchozí pozice blikají coby tečky. V horní části obrazovky roluje průběžná zpráva jak si stojíte vůči zvolenému soupeři.

4. Konec závodu

Váš závod může skončit třemi způsoby:

- a) Diskvalifikací (V kterémkoliv úseku trasy nedodržíte stanovený čas)
- b) Závod dokončíte, ale neumístíte se mezi prvními deseti. Počítač Vám schovívavě gratuluje a vytiskne váš čas pod polem ohněch destei soupeřů. Můžete si tak srovnat časy
- c) Závod ukončíte mezi první desítkou. Počítač vám na příslušném místě vytvoří mezeru pro vypsání vašeho jména a příjmení.

5. Jak nahrát na pásek nové pole deseti mejlepších :

Jestliže splníte podmínky bodu 4c), počítač se Vás zeptá, jestli chcete výsledky uložit na mg.pásek. Jestli ANO - Y. Na obrazovce se objeví nápis:"Insert TOP TEN" tape and press Rec/Play". V případě, že nechcete nahrát potom N ( NO ).

6. Závěr

- chcete-li se opět vrátit kdykoliv na start pak stisněte START
- chcete-li si kdykoliv zobrazit startovací listinu soupeřů - SELECT
- chcete-li přerušit závod - SPACE BAR. Opětovným stiskem budete v závodě pokračovat.

Přeji Vám šťastnou cestu, hodně trpělivosti a umu jménem svým i jménem autora této hry pana Alexe Demea.

Inf. Jaroslav Rádek

**Zpravodaj č. 2 Sekce Olomouc AK**

**Vychází v nákladu 2 000 kusů.**

**Neprodejně – odběr vázán na zaplacení příspěvku 20,- Kčs.**

**Předáno do tisku: 31.3.1987**

**Odpovědný redaktor: Ing. Kopečný Pavel**

**Odborný redaktor: Ing. Pavlík Dobromil**

**Tisk: DRUPOS Bano**

**Tisk povolen: O.K. ONV Olomouc č. o 380500387**