USR V BASICU > ASSEMBLER 23760

```
RANDOMIZE USR (PEEK VAL"23635"+
VAL"256"* PEEK VAL"23636"+ VAL "5")
```

TEST NAHRATIA SUBORU (TR-DOS)

```
LET err= USR VAL"15619":REM :LOAD ""
CODE
IF err<> BIN THEN PRINT "NELZE NAHRAT":
STOP
```

VYPIS 16 BITOVEHO CISLA

```
PRINT PEEK n + 256 * PEEK n+1
```

ZAPIS 16 BITOVEHO CISLA

```
POKE n,v - 256 * INT (v/256)
POKE n+1, INT (v/256)
```

VELKE PISMENA

```
FOR n=64 TO 71: POKE 23681,n: LPRINT
"STOP TAPE": NEXT n
```

VYČISTENIE ATRIBUTOV

```
1 FOR n=22528 TO 23295
2 POKE n,56  (INK 0: PAPER 7)
3 NEXT n
```

INPUT " " KURZOR "C"

```
POKE 23658,8
```

INPUT " " KURZOR "L"

```
POKE 23658,0
```

SAVE BASIC AKO "CODE"

```
SAVE "NO" CODE 23552, CLEAR-23552-
65536+USR 7962: GO TO 1
```

TIME - CAS - FRAMES

```
PRINT (PEEK 23672+ 256* PEEK 23673*
PEEK 23674)/50
```

CLEAR (RAMTOP)

```
PRINT PEEK 23730+ 256* PEEK 23731
PRINT USR 7962-300 (ADRESY 23730-1)
```

VYPIS FREE BASIC MEMORY

```
PRINT 65535 - USR 7962
```

UDG (UZIVATELOM DEFINOVANA GRAFIKA)

```
PRINT PEEK 23675+ 256* PEEK 23676

FOR n=0 TO 71: READ a: POKE USR
"a"+n,a: NEXT n: DATA 0,0,0...
```

START TAPE BEZ CAKANIA

```
POKE 23736,181 (187)
```

NENAPISE "PROGRAM:"

```
POKE 23739,111/CODE "o"
```

NENAPISE "SCROLL?"

```
POKE 23692,0
```

SCROLL O 1 RIADOK

```
RANDOMIZE USR 3280
```

SCROLL O 22 RIADKOV

```
RANDOMIZE USR 3330
```

VYPIS DOBY CAKANIA NA KLAVESU V 1/50 sek.

```
PRINT USR 7997
```

PIPNUTIE

```
RANDOMIZE USR 949
```

Z 128 BASIC DO 48 BASIC

USR 0 so povol. strankovanim Ram
SPECTRUM s zachovanym Basicom

```
10 PLAY "T240N7cccc"
20 PLAY "T60N3cccc"
30 PLAY "T60N3cV15cV15cV15c"
```


nairam
24.08.2002 nairam@nairam.sk

+2A +3DOS & RAMDISK

```
CAT EXP
CAT "5m:" cat user number 0-15
CAT #3 to printer
LOAD"t:" select tape for next load
LOAD "m:*.ba?"
LOAD "a:" bootovatelna disketa
MOVE "*.*" TO "+P" +prot +sys +arc (+ or -)
COPY "a:*.*" TO "m:"
COPY "file" TO SCREEN$
SAVE "5a:file"
POKE 23388,ram_page (ako port #7FFD)
```

BETADISK TR-DOS

```
RETURN
RANDOMIZE USR 15616
RANDOMIZE USR 15619: REM:
LOAD "b:file"
RUN "file"
LIST
CAT
FORMAT "disk"
COPY "A:file", "b:file"
*"b:"
GO
NEW "old","new"
!   ! ! run Devast monitor in ROM
```

MDOS D40 / D80

```
LIST *  INFO O SYSTEME
LOAD *"run"
CAT
CAT "b:"
MOVE "diskA:file","diskB"
LET FN("old.B")="new.B"
LOAD *"run.p"  >>.RUN
```

+3e (G. Lancaster)

```
FORMAT TO 0,42,16,63,10
CAT TAB  info o IDE
NEW EXP "meno",3  swap partition 3MB
NEW DATA "part",3  nova partition 3MB
MOVE "c:" IN "part" ASN
MOVE "part" BIN  erase partition
CAT ASN  info o drajvoch
LOAD "c:" ASN  boot z C: napevno aj po resete
MOVE "c:" OUT ASN  zrusi C:
```

+2 (gray) 128K RAMDISK

```
LOAD !"file"
SAVE !"file"
CAT!
```

BS-DOS MB-02

```
@  = nastavenie disku 9=ramdisk
$  = nastavenie adresara
%  = hexadecimalne cisla v BSROM
↑  = RANDOMIZE USR 0
```

+2A +3 ROM out ports

1FFD	7FFD	ROM
8189	32765	
0	0	0 Editor
0	16	1 Syntax
4	0	2 +3DOS
4	16	3 48BASIC

+2A +3 ALLRAM

OUT	0	16384	32768	49152	65535
8189,5	RAM4	RAM5	RAM6	RAM3	
8189,7	RAM4	RAM7	RAM6	RAM3	
8189,1	RAM0	RAM1	RAM2	RAM3	
8189,3	RAM4	RAM5	RAM6	RAM7	

TEST (IX)==#FFFx

```
ld  a,#0F
or  (IX)
and (IX+1)
inc a
jp  z,nnnn
```

rychly prenos

```
ld  hl,odkial
ld  de,kam
ldi  (hl)  -> (de)  a znizi BC -1
ldi
ldi   x krat
```

Speed copy

```
ld  sp,source

pop hl
ld  (dest1),hl
pop hl
ld  (dest2),hl
jp  z,nnnn
```

Pocitadlo 16bit. BC + FF

```
    ld  bc,#01FF
loop dec bc
     inc b
     djnz loop
```

Vram Driver ;-)

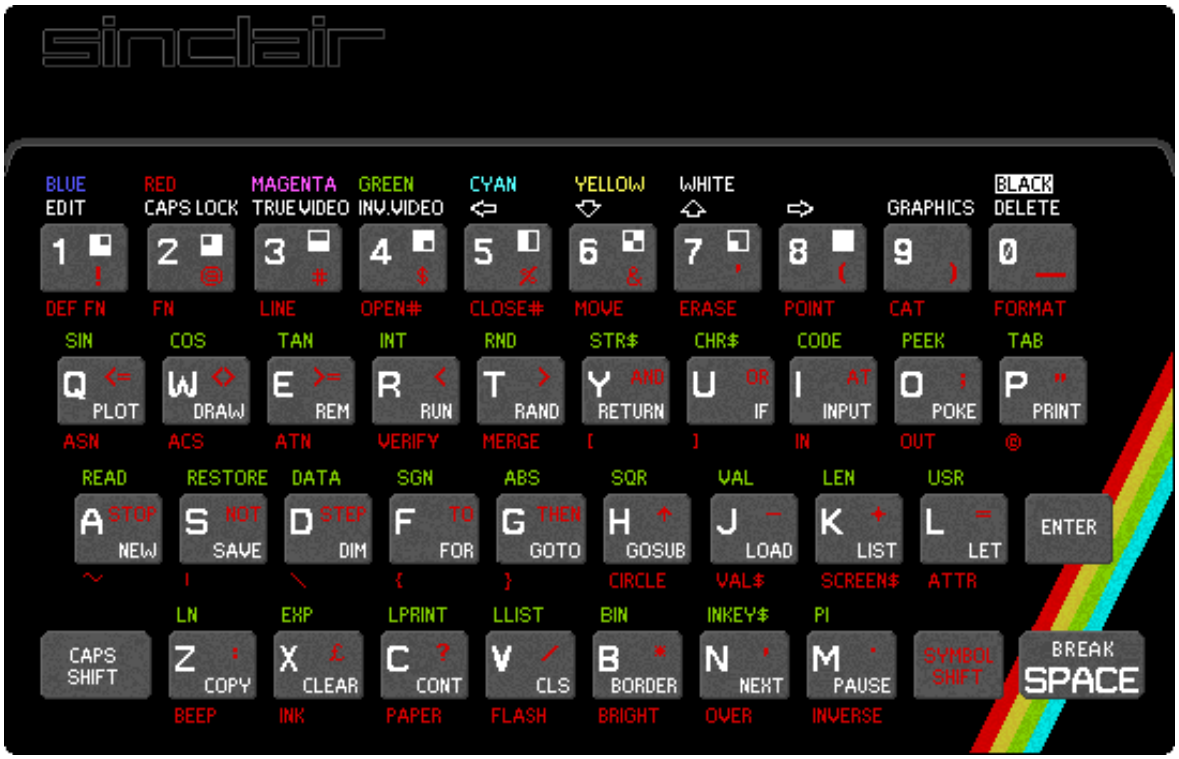
```
ld  sp,source

pop hl
ld  (hl),nn
```

Wait 16bit.dlhe cakanie

```
    ld bc,#FFFF
loop dec bc
     ld a,b
     or c
     jr nz,loop
```

KEYBOARD from IM1

```
nic  halt
     bit 5,(iy+1)
     jr z,nic
     res 5,(iy+1)
     ld a,(23560)
```



COPY all reg. = 20 bytes

```
ld   sp,source
pop  af
pop  bc
pop  de
pop  hl
pop  ix
pop  iy
exx
pop  af
pop  bc
pop  de
pop  hl
ld   sp,destination
push hl
push de
push bc
push af
exx
push iy
push ix
push hl
push de
push bc
push af
```

PRINT RST #10

```
ld   a,2
call #1601
ld   a,"a"  znak
rst  #10
```

DOLE v HL je adr. videoram

```
dole inc  h
     ld   a,h
     and  #07
     ret  nz
     ld   a,#20
     add  a,l
     ld   l,a
     ret  c
     ld   a,#F8
     add  a,h
     ld   h,a
     ret
```

IM2 inicializacia

```
     org  #8000
     di
     ld   a,#80
     ld   i,a
     im2
     ret

     org  #80FF
     dw   run

run  pushaf  //di
     push bc
     pushhl
     ....
     pop  hl
     pop  bc
     pop  af
     jp   #38  keyb
```

```
vynuluje A
xor a

ld  a,page_ram
ld  bc,#7FFD
out (c),a
```

#FB		ZX Printer
#F7 / #EF		Interface 1
#2FFD		read +3FDC
#3FFD		write +3FDC

LDIR + CALL

```
ld   hl,zdroj
ld   de,ciel
ld   bc,dlzka
push de
ldir
ret  //run  ciel
```

8/16bit DEC >> ASCII znak

```
     ld   hl,cislo
d16  ld   de,-10000
     call vypis
     ld   de,-1000
     call vypis
d8   ld   de,-100
     call vypis
     ld   de,-10
     call vypis
     ld   de,-1
vypis ld  a,#FF
vyp1 add  hl,de
     jr   c,vyp1
     and  a
     sbc  hl,de  hl=hl-de-cy(0)
     add  a,#30
     ld   (znak),a
     push hl
     call print
     pop  hl
     ret
```

test ENTER key

```
ld   a,#BF
in   a,(#FE)
rra
ret  nc
```

test BREAK in ROM

```
call #1F54
ret  nc
```

48 TO 128 BASIC — 128 or +2

```
AF          xor a
01 FD 7F    ld bc,#7ffd
ED 79       out (c),a
C3 37 01    jp #0137
```

(DEC: 175,1,253,127,237,121,195,55,1)

48 TO 128 BASIC — +2A or +3

```
AF          xor a
01 FD 7F    ld bc,#7ffd
ED 79       out (c),a
06 1F       ld b,#1f
ED 79       out (c),a
C3 69 01    jp #0169
```

(DEC: 175,1,253,127, 237,121,6,31,237,121, 195,105,1)

KEYBOARD in xxFE (254) 76543210

```
Adresa          Klavesy

FEFE  [65278]  V,C,X,Z,CS
FDFE  [65022]  G,F,D,S,A
FBFE  [64510]  T,R,E,W,Q
F7FE  [63486]  5,4,3,2,1
EFFE  [61438]  6,7,8,9,0
DFFE  [57342]  Y,U,I,O,P
BFFE  [49150]  H,J,K,L,ENTER
7FFE  [32766]  B,N,M,SS,SP
```

```
ld a,XX
in a,(#FE)
```

```
OR-ovane ALL KEYS
in a,(#FE)
```