Zoznam služieb TR-DOS 5.03
LD C, číslo služby
CALL #3D13 (15635)

Číslo služby	Adresa v ROM	Nazov	Popis	Vstupy / výstupy > <
#00	#3D98	RESTORE	Navrat hlavy na stopu 0	<BC:chyba
#01	#3DCB	DRIVE	Nadstavenie mechaniky	>A:číslo mechaniky 0-3 <BC:chyba
#02	#3E63	SEEK	Krokuje na stopu	>A:číslo stopy <BC:chyba
#03	#3F02	SECTOR	Zapíše číslo sektoru	>A:číslo sektoru <BC:chyba
#04	#3F06	BUFFER	Zapíše adresu bufferu	>HL:adresa bufferu <23808 #5D00:adresa <BC:chyba
#05	#1E3D	READ	Načítanie sektoru	>HL:buffer >D:stopa (23797) >E:sektor (23796) >B:počet sektorov (ld bc,#xx05) <BC:chyba
#06	#1E4D	WRITE	Zápis sektoru	>HL:buffer >D:stopa (23797) >E:sektor (23796) >B:počet sektorov (ld bc,#xx06) <BC:chyba
#07	#28D8	CAT	Výpis katalógu	>A:kanál <BC:chyba
#08	#165C	HEADREAD	Načítanie hlavičky	>A:číslo <23773 #5CDD:hlavička <BC:chyba
#09	#1664	HEADWRITE	Zápis hlavičky	>23773 #5CDD:hlavička >A:číslo <BC:chyba
#0A	#1CF0	HEADSEEK	Hľadá hlavičku	>23773 #5CDD:hlavička >23814 #5D06:počet znakov <23823 #5D0F:číslo hlavičky <C:číslo hlavičky #FF=not found
#0B	#28FB	SAVECODE	Zapíše blok typu code	>23773 #5CDD:hlavička >HL:začiatok bloku >DE:dĺžka bloku <BC:chyba
#0C	#28F2	SAVEBASIC	Zapíše blok typu basic	>23773 #5CDD:hlavička >#5CD1:0-neštartuje; iné riadok štartu <BC:chyba
#0D	#01D3	NOT USED		
#0E	#290F	LOAD PRG	Nahrá program	>23773 #5CDD:hlavička <BC:chyba
#0F	#0D13	NOT USED		
#10	#0D13	NOT USED		
#11	#0D13	NOT USED		
#12	#2926	ERASE	Zmazanie súboru	>23773 #5CDD:hlavička <BC:chyba
#13	#28E0	MEM>HEAD	Prenesenie hlavičky z iného miesta na #5CDD	>HL:odkiaľ preniesť <23773 #5CDD:hlavička
#14	#28E0	HEAD>MEM	Prenesenie hlavičky na iné miesto v pamäti	>HL:kam preniesť <(HL):hlavička
#15	#2739	VERIFYTRK	Kontrola stopy čítaním	>D:stopa <23766 #5CD6:počet chýb <BC:chyba
#16	#1FEB	SIDE 0	Volba strany 0	
#17	#1FF6	SIDE 1	Volba strany 1	
#18	#0405	SPECIFY	Nadstavenie premenných podľa stavu mechaniky	<BC:chyba

+3DOS

Adresa v ROM	Nazov	>/< vstup/vystup
0100h (256)	DOS INITIALISE	< A=error code
0103h (259)	DOS VERSION	< D=issue E=version
0106h (262)	DOS OPEN	> B=file number(0-15) C=acces mode (1=read 2=write 3=readwrite 5=sharedread) D=create E=open HL=addres of filename 0=Error File already exists. 1=Open file+header 2=Open file(0) 3=filename.type 4=Erase exist version 0=Error File does not exist 1=Create+open+header 2=Create+open(0)
0109h (265)	DOS CLOSE	> B=file number < A=error code If_OK(CARRY=1)
010Ch (268)	DOS ABANDON	> B=file number If_OK(CARRY=1)
010Fh (271)	DOS REF HEAD	> B=file number If_OK(C=1 Z=1 IX=address of header data in ram7)
0112h (274)	DOS READ	> B=file number C=page DE=lenght HL=address to be read
0115h (277)	DOS WRITE	> B=file number C=page DE=lenght HL=address to be read
0118h (280)	DOS BYTE READ	> B=file number Tests for soft-EOF (1Ah (26)).
011Bh (283)	DOS BYTE WRITE	> B=file number C=byte to write
011Eh (286)	DOS CATALOG	> B=n+1 size of buffer C=filter DE=address buffer HL=address filename
0121h (289)	DOS FREE SPACE	> A=drive ascii 'A'...'P'
0127h (295)	DOS RENAME	> DE=address new filename HL=address old filename
012Ah (298)	DOS BOOT	routine loads a single bootstrap sector A: rampage=4,7,6,3
012Dh (301)	DOS SET DRIVE	> A=drive ascii 'A'...'P' (FFh (255) = get default drive)
0130h (304)	DOS SET USER	> A=User 0...15 (FFh (255) = get default user)
0133h (307)	DOS GET POSITION	> B=file number < C=1, E HL = File pointer 000000h...FFFFFFh (0...16777215)
0136h (310)	DOS SET POSITION	> B=file numbers, E HL = File pointer 000000h...FFFFFFh (0...16777215)
0139h (313)	DOS GET EOF	> B=file number
013Ch (316)	DOS GET 1346	< D=first buffer of cache E=number of cache sector buffers H = First buffer of RAMdisk L = Number of RAMdisk sector buffers
013Fh (319)	DOS SET 1346	> D=first buffer of cache E=number of cache sector buffers H = First buffer of RAMdisk L = Number of RAMdisk sector buffers (Note that E + L <= 128)
0142h (322)	DOS FLUSH	> A=drive ascii 'A'...'P'
0145h (325)	DOS SET ACCESS	> B=file number C=access mode(1=read 2=write 3=readwrite 5=shared)
0148h (328)	DOS SET ATTRIBUTES	> D=attributes set E=attributes clear HL=address of filename
014Bh (331)	DOS OPEN DRIVE	> A=drive B=filenumber C=access mode(1=read 2=write 3=readwrite 5=shared)
014Eh (334)	DOS SET MESSAGE	> A=0(off) FF(on) HL=address of alert routine < HL=address of previous alert routine
0151h (337)	DOS REF XDPB	> A=drive ascii 'A'...'P'
0154h (340)	DOS MAP B	C=unit 0/1 HL=address of change disk routine if unit=0
0157h (343)	DD INTERFACE	floppy disk drive interface present? < C=1(present)
015Ah (346)	DD INIT	Initialise the disk driver
015Dh (349)	DD SETUP	> HL = Address of parameter block (byte:0=motorOnTime 1=motorOffTime 2=writeOffTime 4=StepRate 5=HeadUnloadTime 6=HeadLoadTime)
0160h (352)	DD SET RETRY	> A = Try/retry count >= 1
0163h (355)	DD READ SECTOR	> B=rampage C=unit0/1 D=logicalTrack E=logicalSector HL=address buffer IX=address XDPB
0166h (358)	DD WRITE SECTOR	> B=rampage C=unit0/1 D=logicalTrack E=logicalSector HL=address buffer IX=address XDPB
0169h (361)	DD CHECK SECTOR	> B=rampage C=unit0/1 D=logicalTrack E=logicalSector HL=address buffer IX=address XDPB
016Ch (364)	DD FORMAT	> B=rampage C=unit0/1 D=logicalTrack E=logicalSector HL=address buffer IX=address XDPB Buffer(C=track(0-39) H=headNumber(0=SS) R=sector(0-FF) N=sec.size(2-512B))
016Fh (367)	DD READ ID	> C=unit0/1 D=logicalTrack IX=address XDPB < A=sector number from ID
0172h (370)	DD TEST UNSUITABLE	> C=unit0/1 IX=address XDPB
0175h (373)	DD LOGIN INI XDPB	> C=unit0/1 IX=address XDPB < A=diskType DE=size allocation vector HL=size of hash table
0178h (376)	DD SEL FORMAT	> A=diskType (0=+3 1=Amstrad system 2=Amstrad dataOnly 3=Amstrad PCW) IX=address XDPB < A=diskType DE=size allocation vector HL=size of hash table
017Bh (379)	DD ASK 1	Check unit 1 present. Motor=On
017Eh (382)	DD DRIVE STATUS	> C=unit/head(bit0..1=unit,bit2=head,bit3..7=0) < A = ST3 (Status register 3 of uPD765A)
0181h (385)	DD EQUIPMENT	> C=unit0/1 IX=address XDPB < A=(bit0=SS,1=DS,2=SD,3=DD)
0184h (388)	DD ENCODE	> copy protect A=(FF=On 0=Off) HL=address encode subroutine
0187h (391)	DD L XDPB	> IX=address destination XDPB HL=addr. source disk specification < A=diskType DE=size allocation vector HL=size of hash table
018Ah (394)	DD L DPB	> IX=address destination DPB HL=addr. source disk specification < A=diskType DE=size allocation vector HL=size of hash table
018Dh (397)	DD L SEEK	> C=unit/head(bit0..1=unit,bit2=head,bit3..7=0) D=Track IX=address XDPB
0190h (400)	DD L READ	> HL=addr.param block(byte0=page,1-2=buffer,3-4=lenght,5=numberCommand,6=Command) Low level uPD765A read command
0193h (403)	DD L WRITE	> HL=addr.param block(byte0=page,1-2=buffer,3-4=lenght,5=numberCommand,6=Command) Low level uPD765A write command
0196h (406)	DD L ON MOTOR	Wait for the motor on time as set by DD SETUP
0199h (409)	DD L T OFF MOTOR	Start the motor off time-out
019Ch (412)	DD L OFF MOTOR	Motor Off

BSDOS

Nastrankuje BS-DOS RAM **CALL #3C9E**
Nastrankuje naspat ROM48K **CALL #3C9C**
call #3CA2 TAPE
call #3CA4 DISK

Sluzby BS-**DOS**
LD A,cislo sluzby
RST #20

cislo sluzby	nazov	>/< vstupy/vystupy
#00	GETVER	< BC=ver. DE=addr 1K buff ad. sec HL=addr 1K vseob. IX=addr.256buff re boot IY=addr 4K buff. FAT
#10	KUKNI	Aktualizacia CACHE
#11	KUINI	
#12	GETCST	< D=aktualny adresar, E=aktualny disk, HL=load-kurzor
#13	SETDRV	> E=cislo mechaniky
#14	SETDIR	> E=cislo adresara
#15	SETLCX	> BC=load-kurzor
#16	GETSUB	> BC=cislo suboru (load-kurzor), IX=adresa 32-bajtoveho buffera
#18	KOLKOX	< DE=pocet adr. polozlek v akt. adresari
#19	SEASX	> HL=adr. 10-zn. mena, E=typ suboru (hladanie suboru)
#1A	SEADX	> HL=adr. 10-zn. mena (hlad. adresara)
#1B	OWSUB	> HL=cislo suboru, IX=adresa 32-bajtoveho buffera (Prepisanie adresarovej polozky)
#1E	KILLX	Zmazanie vnutornych CACHE
#1F	FREEX	< DEHL=volne miesto v bajtoch, BC=volne miesto v sektoroch
#22	LOA1X	> HL=cislo suboru, IX=adresa pameti, BC=zaciatocny sektor, E=pocet sektorov
#23	LOA2X	nahrava cely sektor
#24	VERIX	
#25	SAVEX	
#26	CREATX	> IX=adr. 32-bajtovej adresarovej polozky < HL=cislo vytvoreneho suboru
#27	UTRASX	Utrasenie aktualneho adresara
#28	ERASEX	> BC=cislo suboru
#29	ERASIX	> BC..HL=interval suborov
#2A	MOVEXX	> BC=cislo suboru, E=cielovy adresar
#2B	MOVEIX	> BC..HL=interval suborov, E=cielovy adresar
#2C	CATX	> C=cislo adresara ktory vypise (B=1 aktualny adresar)
#2D	SEAXXX	> BC=dlzka retazca (max 10), DE=adr. retazca (hladanie)
#2E	CATXD	> BC=specifikacia adresara, L=cislo mechaniky
#2D	SEAXXD	> BC=dlzka retazca (max 10), DE=adresa retazca, L=cislo mechaniky
#30	START	Start operacneho systemu
#33	FORMX	Rychly format diskety
#34	MESSX	> HL=addr. textu < C=enter NC=space
#36	UNERAX	> BC=cislo suboru (unerase) BC=0(cely adresar)

Sluzby BS-**BIOS**
LD A,cislo sluzby
RST #18

cislo sluzby	nazov	vstupy/vystupy
#00	RESET	Resetuje FDC kontroler
#01	STATUS	< A=status
#02	RESTORE	Hlavicka na 0 stopu
#03	SEEK	> D=stopa
#04	RDSEC	> DE=sektor, HL=adresa #05WRSEC> DE=sektor, HL=adresa
#06	RDTRK	> D=stopa, bit 7,E=strana, HL=adresa
#07	WRTRK	> D=stopa, bit 7,E=strana, HL=adresa
#08	RDADD	> D=stopa, bit 7,E=strana, HL=adresa
#09	REZIM	> E=0 pre DD, 1 pre HD, 2 pre ED < Z=rezim nastaveny, NZ=error
#0C	AKTIVE	> E=cislo mechaniky < A=0Unknown 1NotReady 2NevymenenaOk 3VymenenaOk
#0D	PASIVE	Vypina aktivnu mechaniku
#0E	INFO	> E=cislo mechaniky < D=pocet stop E=rezim
#0F	SETMAX	> BC=max.prenas. dlzka DMA
#10	SETMLT	> HL=addr rutinky
#11	SETOFF	> E=0(motorOff) FF(motorOn)

STATUS A-reg.

Bit	Chyba	Vyznam chyby
0	time out	operacia trvala dlhsie, ako sa patri
1	disk not ready	mechanika nie je pripravena
2	lost data	chyba pri prenose dat disk-pamet
	track 0	hlava je na stope 0
3	CRC error	nesedi kontrolny sucet pri citani
4	record not found	nenasiel sa sektor
	seek error	nepodarilo sa najst nultu stopu
5	break	pri diskovej operacii bolo stlacene QW
6	write protect	disk chraneny proti zapisu
7	disk not ready	mechanika nie je pripravena

nairam
nairam@nairam.sk
19.02.2004

Fyzicke cislo sektora DE:

D	7	6	5	4	3	2	1	0
	cislo stopy							

E	7	6	5	4	3	2	1	0
	strana	cislo sektora						