



Číslo služby	Adresa v ROM	Nazov	Popis	Vstupy / výstupy
#00	#3D98	RESTORE	Navrat hlavy na stopu 0	<BC:chyba
#01	#3DCB	DRIVE	Nadstavienie mechaniky	>A:číslo mechaniky 0-3 <BC:chyba
#02	#3E63	SEEK	Krokuje na stopu	>A:číslo stopy <BC:chyba
#03	#3F02	SECTOR	Zapiše číslo sektoru	>A:číslo sektoru <BC:chyba
#04	#3F06	BUFFER	Zapiše 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	Zapiše blok typu code	>23773 #5CDD:hlavička >HL:začiatok bloku >DE:dlžka bloku <BC:chyba
#0C	#28F2	SAVEBASIC	Zapiše 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	Nadstavienie premenných podľa stavu mechaniky	<BC:chyba

Adresa v ROM	Nazov	Popis
0100h (256)	DOS INITIALISE	
0103h (259)	DOS VERSION	
0106h (262)	DOS OPEN	
0109h (265)	DOS CLOSE	
010Ch (268)	DOS ABANDON	
010Fh (271)	DOS REF HEAD	
0112h (274)	DOS READ	
0115h (277)	DOS WRITE	
0118h (280)	DOS BYTE READ	
011Bh (283)	DOS BYTE WRITE	
011Eh (286)	DOS CATALOG	
0121h (289)	DOS FREE SPACE	
0127h (295)	DOS RENAME	
012Ah (298)	DOS BOOT	
012Dh (301)	DOS SET DRIVE	
0130h (304)	DOS SET USER	
0133h (307)	DOS GET POSITION	
0136h (310)	DOS SET POSITION	
0139h (313)	DOS GET EOF	
013Ch (316)	DOS GET 1346	
013Fh (319)	DOS SET 1346	
0142h (322)	DOS FLUSH	
0145h (325)	DOS SET ACCESS	
0148h (328)	DOS SET ATTRIBUTES	
014Bh (331)	DOS OPEN DRIVE	
014Eh (334)	DOS SET MESSAGE	
0151h (337)	DOS REF XDPB	
0154h (340)	DOS MAP B	
0157h (343)	DD INTERFACE	
015Ah (346)	DD INIT	
015Dh (349)	DD SETUP	
0160h (352)	DD SET RETRY	
0163h (355)	DD READ SECTOR	
0166h (358)	DD WRITE SECTOR	
0169h (361)	DD CHECK SECTOR	
016Ch (364)	DD FORMAT	
016Fh (367)	DD READ ID	
0172h (370)	DD TEST UNSUITABLE	
0175h (373)	DD LOGIN INI XDPB	
0178h (376)	DD SEL FORMAT	
017Bh (379)	DD ASK 1	
017Eh (382)	DD DRIVE STATUS	
0181h (385)	DD EQUIPMENT	
0184h (388)	DD ENCODE	
0187h (391)	DD L XDPB	
018Ah (394)	DD L DPB	
018Dh (397)	DD L SEEK	
0190h (400)	DD L READ	
0193h (403)	DD L WRITE	
0196h (406)	DD L ON MOTOR	
0199h (409)	DD L T OFF MOTOR	
019Ch (412)	DD L OFF MOTOR	

>< vstup/vystup
<A=error code
<D=issue E=version
>B=file number(0-15) C=access mode (1=read 2=write 3=readwrite 5=shared) D=create E=open HL=address of filename 0=Error File already exists. 1=Open file+header 2=Open file(0) 3=filename.type 4=Erased exist version 0=Error File does not exist 1=Create+open+header 2=Create+open(0)
>B=file number
<A=error code If_OK(CARRY=1)
>B=file number If_OK(CARRY=1)
>B=file number If_OK(C=1 Z=1 IX=address of header data in ram7)
>B=file number C=page DE=length HL=address to be read
>B=file number C=page DE=length HL=address to be read
>B=file number Tests for soft-EOF (1Ah) (26h).
>B=file number C=byte to write
>B=n+1 size of buffer C=filter DE=address buffer HL=address filename
>A=drive ascii 'A...' 'P'
>DE=address new filename HL=address old filename
routine loads a single bootstrap sector A: rampage=4,7,6,3
>A=drive ascii 'A...' 'P' (FFh) (255) = get default drive)
>A=User 0...15 (FFh) (255) = get default user)
>B=file number
<C=1, E HL = File pointer 000000h...FFFFFFh (0...16777215)
>B=file numbers, E HL = File pointer 000000h...FFFFFFh (0...16777215)
>B=file number
<D=first buffer of cache E=number of cache sector buffers H = First buffer of RAMdisk L = Number of RAMdisk sector buffers
>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)
>A=drive ascii 'A...' 'P'
>B=file number C=access mode(1=read 2=write 3=readwrite 5=shared)
>D=attributes set E=attributes clear HL=address of filename
>A=drive B=file number C=access mode(1=read 2=write 3=readwrite 5=shared)
>A=(off) FF(on) HL=address of alert routine
<HL=address of previous alert routine
>A=drive ascii 'A...' 'P'
C=unit 0/1 HL=address of change disk routine if unit=0
floppy disk drive interface present? < C=1(present)
Initialise the disk driver
> HL = Address of parameter block (byte:0=motorOnTime 1=motorOffTime 2=writeOffTime 4=StepRate 5=HeadUnloadTime 6=HeadLoadTime)
> A = Try/retry count >= 1
> B=rampage C=unit0/1 D=logicalTrack E=logicalSector HL=address buffer IX=address XDPB
> B=rampage C=unit0/1 D=logicalTrack E=logicalSector HL=address buffer IX=address XDPB
> 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))
> C=unit0/1 D=logicalTrack IX=address XDPB
<A=sector number from ID
> C=unit0/1 IX=address XDPB
> C=unit0/1 IX=address XDPB
<A=diskType DE=size allocation vector HL=size of hash table
>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
Check unit 1 present. Motor=On
> C=unit/head(bit0..1=unit,bit2=head,bit3..7=0)
<A = ST3 (Status register 3 of uPD765A)
> C=unit0/1 IX=address XDPB
<A=(bit0=SS,1=DS,2=SD,3=DD)
> copy protect A=(FF=On 0=Off) HL=address encode subroutine
> IX=address destination XDPB HL=addr. source disk specification
<A=diskType DE=size allocation vector HL=size of hash table
> IX=address destination DPB HL=addr. source disk specification
<A=diskType DE=size allocation vector HL=size of hash table
> C=unit/head(bit0..1=unit,bit2=head,bit3..7=0) D=Track IX=address XDPB
> HL=addr.param block(byte0=page,1-2=buffer,3-4=length,5=numberCommand,6=Command) Low level uPD765A read command
> HL=addr.param block(byte0=page,1-2=buffer,3-4=length,5=numberCommand,6=Command) Low level uPD765A write command
Wait for the motor on time as set by DD SETUP
Start the motor off time-out
Motor Off

+3DOS

Nastrankuje BS-DOS RAM CALL #3C9E	Nastrankuje naspas ROM48K CALL #3C9C	call #3CA2 TAPE call #3CA4 DISK	BSDOS
číslo služby	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	Služby BS-DOS LD A, číslo služby RST #20
#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. poloziek 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	FREEEX	< 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=specificacia adresara, L=cislo mechaniky	
#2D	SEAXXD	> BC=dlzka retazca (max 10), DE=adresa retazca, L=cislo mechaniky	
#30	START	Start pracovneho 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)	

číslo služby	nazov	vstupy/vystupy	Služby BS-BIOS LD A, číslo služby RST #18
#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	RDRTRK	> 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 2NvymenenaOk 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
2	track 0	hlava je na stope 0
3	CRC error	neseadi kontrolny sučet pri citani
4	record not found	nenasiel sa sektor
4	seek error	nepodarilo sa najst nultu stopu
5	break	pri diskovej operaci bolo stlacene QW
6	write protect	disk chranyeny proti zapisu
7	disk not ready	mechanika nie je pripravena



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								