Micro Vibe

shaels.net/index.php/cpm80-22-documents/cpm-bdos/31-bdos-overview

My Micro Hobby Projects

BDOS Overview

Tuesday, 08 December 2009 07:39 Shael

The BDOS portion of CPM2.2 allows programs to access the services provided by the CPM operating system. The BDOS provided thirty eight service functions for programs running under CPM2.2. A program accesses the BDOS using an 8080 jump instruction located at address 0005 in memory. To request a BDOS service the program uses an 8080 CALL instruction to memory address 0005. On entry to the call the CPU "C" register contains the BDOS service request number and the "DE" CPU register is used to pass a 16 bit parameter or the "E" register is used for an 8 bit parameter. When the call returns to the program results will be passed to the program in a CPU registers "A" and/or "HL". Many of the service functions relating to file service and file IO require the memory address of an FCB (File Control Block) structure.
I have grouped the services provided by the BDOS into the following categories; system services, serial device services, file services, file I/O services, and disk device services.

The tables below provide a quick reference to the CPM2.2 BDOS services broken down by category,

System Services

Service	Description	Parameters	Result
0	Terminate program	n/a	n/a
12	Get system version	n/a	H = ident L = version

Serial Services

Service	Description	Parameters	Result

1	Console (CON) read byte with echo	n/a	A = value
2	Console (CON) write byte	E = value	n/a
3	Reader (RDR) read byte	n/a	A = value
4	Punch (PUN) write byte	E = value	n/a
5	List (LST) write byte	E = value	n/a
6	Direct console (CON) read/write	E = 255 read E 255 write	Read A = value or 0 for no data
7	Get IOBYTE	n/a	A = IOBYTE
8	Set IOBYTE	E = value	n/a
9	Console (CON) write "$" terminated 7 bit string	DE = address of string	n/a
10	Console (CON) read line into buffer. First byte max size, second byte length read. Third byte first byte of string.	DE = buffer address	n/a
11	Console (CON) check if byte can be read	n/a	A = 0 not ready

File Services

Service	Description	Parameters	Result
15	Open existing file referenced by FCB	DE = FCB address	A = 0..3 Ok A = 255 failed
16	Close file open for output referenced by FCB	DE = FCB address	A = 0..3 Ok A = 255 failed
17	Search for first file match referenced by FCB	DE = FCB address	A = 0..3 Ok A = 255 failed
18	Find next file match referenced by FCB	DE = FCB address	A = 0..3 Ok A = 255 failed

19	Delete File referenced by FCB	DE = FCB address	A = 0..3 Ok A = 255 failed
22	Make and open new file referenced by FCB	DE = FCB address	A = 0..3 Ok A = 255 failed
23	Rename file, data area of FCB contains new name	DE = FCB address	A = 0..3 Ok A = 255 failed
30	Set file attributes FCB contains new attributes	DE = FCB address	n/a
32	Get / Set the User Code Area	E = 255 Get E = 0..15 Set	A = User

File I/O Services

Service	Description	Parameters	Result
20	Sequential read file referenced by FCB	DE = FCB address	A = 0 Ok A 0 failed A = 255 EOF
21	Sequential write file referenced by FCB	DE = FCB address	A = 0 Ok A = 1 no dir A = 2 full disk
26	Set file transfer buffer address	DE = buffer address	n/a
33	Direct read file	DE = FCB address	A = 0 Ok A 0 failed
34	Direct write file	DE = FCB address	A = 0 Ok A 0 failed
35	Get file end address	DE = FCB address	A = Ok A 0 failed
36	Get direct address	DE = FCB address	A = Ok A 0 failed
40	Direct write file zero fill	DE = FCB address	A = Ok A 0 failed

Disk Services

Service	Description	Parameters	Result

13	Reset all disks	n/a	n/a
14	Select Drive	E = drive	n/a
24	Get active drive map	n/a	HL = one bit for each of 16 drives
25	Get default drive number	n/a	A = drive
27	Get allocation vector	n/a	HL = address of drive allocation vector
28	Protect current drive Make drive read only	n/a	n/a
29	Get read only map	n/a	HL = one bit for each of 16 drives
31	Get active disk disk parameter block (DPB)	n/a	HL = address of DPB

For more information about a group of service functions select the table name link for the service group desired.

Last Updated on Saturday, 16 March 2013 06:00