Appendix

B

DOS Function Calls: INT 21H*

Table B.1

Function Value in AX: AH AL	*Function*	*Register I/P*	*Return O/P*
1. Function 00H (0)	Restore termination handler vector from psp: 000AH	AH = 00, CS = Segment address of program segment prefix	Nothing
Program terminate	Restore the ctrl C -vector from psp : 000EH		
2. Function 01H (1) Character I/P with echo.	Inputs a character from keyboard, then echoes it to display. If no character is ready, waits until one is available.	AH = 01	AL = 8-bit character
3. Function 02H(2) Character output	Output a character to the currently active video display.	AH = 02, DL = 8 bit char. (ASCII code)	Nothing
4. Function 03H(3) Auxiliary input	Reads a character from the first serial port.	AH = 03	AL = 8-bit char.
5. Function 04H(4) Auxiliary output	Output a character to the first serial port.	AH = 04 DL = 8-bit char.	Nothing
6. Function 05H(5) Printer output	Sends a character to the first device (PRN or LPT 1).	AH = 05 DL = 8-bit char.	

(Contd.)

* Compiled by Amesha Thaker B. E. (Comp. Tech.)

Function Value in AX/AH/AL	Function	Register I/P	Return O/P
7. Function 06H(6) Direct console I/O	Reads a character form keyboard or returns zero if none is ready or writes a character to the display	AH = 06, DL = Function requested	If zero flag = clear AL = 8-bit data, else zeroflag = set
8. Function 07H(H) unfiltered char I/P without zero	Reads a character from keyboard without echoing it to the display. If no character is ready, waits until one is available.	AH = 07	AL = 8-bit char
9. Function 08H(8) Char. I/P without echo.	Reads a character from keyboard without echoing it to the display. If no character is ready, waits until one is available.	AH = 08	AL = 8-bit char.
10. Function 09H(9) Output char. string	Sends a string of characters to the display.	AH = 09, DS: DX = Segment: offset of string	–
11. Function 0AH(10) Buffered input	Reads a string of characters from keyboard and places it in a user–designated buffer.	AH = 0AH, DS: DX = Segment: offset of buffer.	–
12. Function 0BH(11) Get input status	Checks whether a character is available from the keyboard.	AH = 0BH	AL = 00, not available = FFH, available
13. Function 0CH(12) Reset I/P buffer and then input	Clears the type ahead buffer and then invokes one of the keyboard input functions.	AH = 0CH, AL = No. of I/p functions to be after reseting invoked buffer: 01H, 06H, 07H, 08H or 0AH. DS:DX = seg.: offset of I/p buffer	If function is 01H, 06H, 07H, 08H, AL is 8-bit data.
14. Function 0DH(13) Disk reset	Selects drive A as the default, set the disk transfer (DTA) address to DS:0080H, and flushes all file buffers to disk.	AH = 0DH	
15. Function 0EH(14) set default disk drive	Selects a specified drive to be the current, or default, disk drive, and returns the total no. of logical drives in system.	AH = 0EH, DL = Drive code (0 = A, 1 = B ...)	AL = No. of logical drives in system.

(Contd.)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
16. Function 0FH(15) Open File	Opens a file and makes it available for subsequent read/write operation	AH = 0FH, DS:DX = segment: offset of file control block	In successful AL = 0 In failed, AL = 0FFH
17. Function 10H(16) Close File	Closes a file, and updates the disk directory if the file has been modified or extended	AH = 10H, DS:DX = segment: offset of File control block	AL = 00-In successful = 0FFH
18. Function 11H(17) Search for first match	Searches current directory on disk in the designated drive for a matching filename	AH = 11H, DS:DX = segment: offset of FCB	If file found, AL = 00 File not found, AL = 0FFH
19. Function 12H(18) Search for next match	Given that a previous call to function 11H has been successful, returns next matching filename (if any)	AH = 12H, DS:DX = segment: offset of FCB	File found, AL = 00 Not found, AL = 0FFH
20. Function 13H(19) Delete file	Deletes all matching files from the current subdirectory	AH = 13H, DS:DX = segment: offset of FCB	File found, AL = 00 File not found, AL = 0FFH
21. Function 14H(20) Sequential read	Reads the next sequential block of data from a file, then increments the file pointer appropriately	AH = 14H, DS:DX = segment: offset of previously opend FCB	AL = 00 if read, 01 if EOF 02 if seg wrap, 03 if partial record read at EOF
22. Function 15H(21) Sequential write	Writes the next sequential block of data into a file, then increments file	AH = 15H, DS:DX = segment: offset of previously opened FCB	AL = 00 if write ok AL = 01 if disk full AL = 02 if seg wrap
23. Function 16H(22) Create or truncate file.	Creates new directory entry in current subdirectory or truncates any existing file with specified length.	AH = 16H, DS:DX = segment: offset of unopened FCB	AL = 00 file created AL = 0FFH file not created
24. Function 17H(23) Rename file	Alters the name of all watching files in current subdirectory on disk in the specified drive.	AH = 17 H, DS:DX = segment: offset of special FCB	AL = 00 if renamed AL = 0FFH if not found
25. Function 18H (24) 26. Function 19H (25) Get default disk drive	Returns drive code of current or default disk drive.	AH = 19H	AL = Drive code

(Contd.)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
27. Function 1A (26) Set DTA address	Specifies memory address to be used for subsequent PCB disk operation.	AH = 1AH, DS:DX = segment: offset of disk transfer area.	
28. Function 1BH(27) Get allocation information for default drive.	The address returned in DS–BX points to the actual FAT.	AL = number of sector per cluster DS:BX = segment: offset of FAT identification byte, CX = Size of physical sector (in bytes), DX = number of clusters for default drive.	AH = 1BH
29. Function 1CH(28) Get allocation information for specified drive.	Obtains selected information about the specified disk drive and a pointer to the identification byte from its file allocation table (FAT)	AL = number of sector per cluster, DS:BX = segment: offset of FAT identification table, CX = Size of physical sector (in bytes), DX = number of clusters for default or specified drive.	AH = 1CH DL = drive code (0 = default, 1 = A, etc.)
30. Function 1DH (29)	Reserved	–	–
31. Function 1EH (30)	Reserved	–	–
32. Function 1FH (31)	Reserved	–	–
33. Function 20H (32)	Reserved	–	–
34. Function 21H (33) Random read	Read a selected record from a file into memory	AH = 21H DS:DX = segment: offset of previously opened file control block.	AL = 00 if read successfully, 01 if end of file, 02 if segment wrap, 03 if partial record read at end of file.
35. Function 22H (34) Random write	Writes data from memory into a selected record in a file.	AH = 22H, DS:DX = segment : offset of previously opened file control block.	AL = 00 if write successfully, 01 if disk full, 02 if segment wrap.
36. Function 23H (35) Get file size in records	Searches for a matching file in the current subdirectory, if one is found, fills a file control block (FCB) with file size information in terms of record count.	AH = 23H, DS:DX = segment: offset of unopened file control block.	If matching file found AL = 0, if not then AL = 0FFH.

(*Contd.*)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
37. Function 24H (36) Set random record number	Sets the random record field of a file control block (FCB) to correspond to the current file position as recorded in the opened FCB.	AH = 24H, DS:DX = segment: offset of previously opened file control block.	Register contents not affected. Random-record field is modified in file control block.
38. Function 25H (37) Set interrupt vector	Initialize a machine interrupt vector to point to an interrupt handling routine.	AH = 25H, AL = machine interrupt number, DS:DX = segment: offset of interrupt handling routine.	Nothing
39. Function 26H (38) Create program segment prefix	Copies the program segment perfix (PSP) of the current executing program to a specified segment address in free memory, then updates the new PSP to make it usable by another program.	AH = 26H DX = segment of new program segment prefix.	Nothing
40. Function 27H (39) Random block read	Reads one or more sequential records from a file into memory, starting at a designated file location.	AH = 27H CX = number of records to be read, DS:DX = segment:offset of previously opened file control block.	AL = 00 if all requested records read, 01 if end of file, 02 if segment wrap, 03 if partial record read at end of file, CX = actual number of records read.
41. Function 28H (40) Random block write	Writes one or more sequential records from a memory to a file, starting at a designated file location.	AH = 28 H CX = number of records to be written, DS:DX = segment: offset of previously opened FCB	AL = 00 if all requested records are written, 01 if disk full, 02 if segment wrap, CX = actual number of records written.
42. Function 29H (41) Parse filename	Parses a text string into the various fields of the control block.	AH = 29 H, AL = flags to control passing, DS:SI = seg.: offset of text string, ES:DI = segment: string, offset, of file control block.	AL = 00 if no global character encountered, 01 if parsed string contain global character, 0FFH if drive specifier invalid, DS:SI = seg.: offset of 1st char. after parsed filename, ES:DI = seg. : offset of formatted unopened file block.

(*Contd.*)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
43. Function 2AH (42) Get system date	Obtains the system day of month, day of the week, month, and year.	AH = 2AH	CX = year (1980-2099) DX = month DL = day
44. Function 2BH (43) Set system date	Initialises system—clock driver to a specified date. The system time is not affected.	AH = 2BH CX = year DH = month DL = day	AL = 00 if date set successfully. = 0FFH if date not valid.
45. Function 2CH (44) Get system time	Obtains time of day from system real time clock driver, converted to hour, minutes, seconds and hundredths of seconds.	AH = 2CH	CH–hour CL–minutes DH–seconds DL–1/100th of secs.
46. Function 2DH (45) Set system time	Initialises system real—time clock to a specified hour, min, sec. and hundredth of second. System date is not affected.	AH = 2DH CH = hours CL = minutes DH = seconds DL = 1/100th of secs.	AL = 00 if time set successfully = 0FFH if not valid.
47. Function 2EH (46) Set verify flag	Turns off or turns on o.s. flag for automatic read–after write verification of data.	AH = 2EH AL = 00	
48. Function 2FH (47) Get disk transfer are a addr.	Obtains current address of DTA for FCB file read/write operation.	AH = 2FH	ES:BX = seg.:offset of DTA
49. Function 30H (48) Get MS-DOS version number	Returns version no. of operating system.	AH = 30H no. no.	AL = major version AH = minor version (3.10 = 0AH(10), etc.)
50. Function 31H (49) Terminate and stay resident (KEEP process)	Terminates a process without releasing its memory.	AH = 31H AL = return code = mem. size to reserve	–
51. Function 32H (50) Reserved	–	–	–
52. Function 33H (51) Get/set Ctrl–Break flag	Determines current status of os's Ctrl–break or Ctrl–C checking flag.	AH = 33H if getting status of Ctrl–Break flag AL = 00, if setting AL = 01, DL = 00, DL = 01.	DL = 00 if C–B checking off, DL = 01 if Ctrl–Break checking on.

(Contd.)

Function Value in AX/AH Al	Function	Register I/P	Return O/P
53. Function 34H (52) Reserved			
54. Function 35H (53) Get interrupt vector	Obtains address of current interrupt handler routine for specified M/C interrupt	AH = 35H AL = int. no	ES:BX = seg. offset of interrupt handle
55. Function 36H (54) Get free disk space	Obtains selected info. about a disk drive from which the drive's capacity can be calculated.	AH = 36H DL = drive code	If drive valid, AX = sectors/cluster BX = no. of clusters CX = bytes/sectors DX = clusters/drive If specified drive invalid AX = FFFFH
56. Function 37H (55) Reserved			
57. Function 38H (56) Get/Set country	Obtains current-country information.	AH = 38H, AL = 00, DS:DX = seg:offset of buffer for returned information. set	If no error occurs, BX = country code DS:DX Bytes 0–1 = date format 2 = currency symbol 3 = zero 4 = thousand. sep. char 5 = zero 6 = decimal sep. char 7 = zero 8–31 = reserved. If error occurs, CY flag = set AX = error code If no error while setting current country code CY = clear If error occurs CY = AX = error code.
58. Function 39H (57) Create sub-directory	Creates sub-directory using specified drive and path.	AH = 39H, DS:DX = seg. offset of ASCIIZ path specification	If function successful, CY = clear Function failed, CY = set, AX = error code.

(Contd.)

Function Value in AX/AH 4I	Function	Register I/P	Return O/p
59 Function 3AH (58) Delete sub directory	Removes sub-directory using specified disk and path	AH = 3AH, DS:DX = seg : offset of ASCIIZ string	If function successful, CY = clear Function failed, CY = set, AX = error code
60 Function 3BH (59) Set current directory	Sets the current or default directory using specified drive and path.	AH = 3BH DS:DX = seg.: offset of ASCIIZ string	If function successful, CY = clear Function failed, CY = set, AX = error code.
61. Function 3 CH (60) Create or truncate file	Creates a new file in the designated or default directory on the designated or default disk drive. If specified file already exists it is truncated to zero length. The file is opened and a 16-bit token, or handle is returned, which is used by the program for further access to the file.	AH = 3CH CX = file attribute, 00H if normal, 01H if read only, 02H if hidden, 04H if system. DS:DX = seg.: offset of ASCIIZ file.	If function successful, carry flag = clear, AX = file handle. If not successful, Carry flag = set, AX = error code, 3-if path not found 4-if no handle 5-if access denied.
62. Function 3DH (61) Open file	Given an ASCIIZ file specification opens the specified file in the designated or default directory on the designated or default disk drive.	AH = 3DH AL = access mode DS:DX = seg.:offset of ASCIIZ file specification.	If function successful, Carry flag = clear, AX = file handle. If not successful CY flag = set, AX = error code.
63. Function 3EH (62) Close file	Given a file token or handle that was returned by a previous successful open (function 3DH) or create operation, flushes all internal buffer to disk, closes the file, and releases the handle for reuse. If file was modified or extended, the time and date, stamp and the file size are updated in directory entry.	AH = 3EH BX = file handle.	If function successful, Carry flag = clear. If not successful, Carry flag = set, AX = error code, 6-if handle invalid or not open.

(Contd.)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
64. Function 3FH (63) Read file or device	Given a valid file token or handle from a previous successful open or create operation, a buffer address and a length in bytes, transfers data at the current file-pointer from the file into the buffer and then updates the file pointer position.	AH = 3FH BX = file handle, CX = no. of bytes to be read. DS:DX = seg.:offset of buffer area.	If function successful, CY flag = clear, AX = no. of bytes read. If failed, CY flag = set, AX = error code.
65. Function 40H (64) Write to file or device	Given a file token or handle from a previous successful open or create operation, a buffer address and a length in bytes, transfers data from the buffer into the file and updates the file pointer positions.	AH = 40H BX = file handle, CX = no. of bytes to be written, DS:DX = seg.:offset CY flag = set	If function successful, CY flag = clear, AX = no. of bytes written. If fn failed, AX = error code.
66. Function 41H (65) Delete file	Deletes a file from the specified or default disk and directory.	AH = 41H DS:DX = Seg.:offset	If function successful, CY flag = clear. If function failed, CY = set, AX = error code.
67. Function 42H (66) Move file pointer	Sets file pointer location relative to the start of the file, the end of file or current file position. half of offset	AH = 42H AL = method code BX = file handle CX = most significant half of offset DX = least significant part of new ptr.	If function successful. Carry flag = clear. DX-most significant part of new ptr. location, AX-least significant location, 1-if function no. valid 6-if handle invalid.
68. Function 43H (67) Get or set file attributes	Obtains or alters the attributes of a file.	AH = 43H AL = 00H if getting file attribute, 01H if setting, CX = new attribute DS:DX = seg.:offset	If function successful, CY flag = clear. If AL = 00 on call CX-attribute. If function failed, CY flag = set, AX = error code.

(*Contd.*)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
69. Function 44H (68) Device driver control (IOCTL)	Passes control information directly between an application and a device driver	AH = 44H AL = 00H if getting device info 01H- if setting device info 02H-if reading from device control channel to buffer 03H-if writing from buffer to device control channel 04H-same as 02H, but codes using drive no. in BL 05H-same as 03H, but using drive no. in BL 06H-if getting I/p status, 07H-if getting O/p status 08H-if testing whether block device changeable 09H-if testing block device local 0AH -if testing handle local 0BH-if changing sharing retry count.	If function successful CY flag = clear AX = no of bytes transferred AX = value if function code 08H AL -status if function 06H-07H DX-device info if function code 00H
70. Function 45H (69) Duplicate handle	Given a handle for a currently open device or file returns a new handle that refers to the same device or file.	AH = 45H BX-file handle	If function successful, CY flag = clear, AX = new file handle. If function failed, CY flag = set, AX = error code, 4-if no handle 6-if handle invalid.
71. Function 46H (70) Force duplicate of handle	Given two handles makes the second handle refer to the same opened file at the same location as first handle.	AH = 46H BX = first file handle CX = second file handle	If function successful, CY flag = clear. If function failed, CY flag = set, AX = error code, 4-if no handle 6-if handle invalid.
72. Function 74H (71) Get current directory	Obtains an ASCIIZ string that describes the path from the root to currently active directory and name of directory.	AH = 47H DL = drive code DS:SI = seg.:offset of 64-byte scratch buffer	If function successful, CY flag = clear. If function failed, CY flag = set, AX = error code.

(Contd.)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
73. Function 48H (72) Allocate memory	Allocates a block of memory and returns a pointer to the beginning of the allocated area	AH = 48H BX = no of paragraphs of memory needed.	If function successful, CY flag = clear, AX initial seg of allocation block If failed, CY flag = set, AX = error code, BX = size of largest available block.
74. Function 49H (73) Release memory	Release a memory block and makes it available for use by other programs.	AH = 49H ES = seg. or block to be released.	If function successful, Carry flag = clear. It not successful, Carry flag = set, AX = error code, 7-if MCB's destroyed 9-if incorrect segment in ES.
75. Function 4AH (74) Modify memory allocation	Dynamically shrinks or extends a memory block according to the needs of an application program.	AH = 4AH BX = new requested block size in paragraphs ES = seg. block to be modified.	If function successful, Carry flag = clear. If not successful, Carry flag = set, AX = error code, BX = max. block size available.
76. Function 4BH (75) Execute program	Allows an application program to run another program, regaining control when it is finished and optionally examining the child program's return code. Can also be used to load overlays, but this use is uncommon.	AH = 4BH AL = 00 if loading + executing program = 03 if loading overlay ES:BX = seg.:offset of parameter block. DS:DX = seg.:offset of prog. specification (file name)	If function successful, Carry flag = clear, All registers except CS and IP are destroyed including SP. If function failed, Carry flag = set, AX = error code, 1-if function invalid 2-if file not found or path invalid 3-if insufficient memory to load the program 5-if access denied 0AH-if environment invalid 0BH-if format invalid.

(Contd.)

Function Value in AX/AH/AL	Function	Register I/P	Return O/P
77. Function 4CH (76) Terminate with return code	Performs a final exit to MS-DOS or to a parent task, passing back a return code. DOS then takes following actions 1. Restores the termination handler vector from PSP:000AH 2. Restores the Ctrl-Break vector from PSP:000EH 3. Restores critical error handler vector from PSP:0012H 4. Flushes file buffers 5. Transfer to termination handler address.	AH = 4CH AL = return code	
78. Function 4DH	Gets return code of child program after its termination.	AH = 4DH	AH = 00-normal termination with function 4CH 01-termination via ^C, INT23H 02-termination due to critical error 03-termination via function 31H. AL = Return code specified while terminating using 4CH or 31H
79. Function 4EH	Search directory for first matching file and report information about it	AH = 4EH, CX = Search attribute, DS:DX = Segment: Offset address of null terminated ASCII string of the filename path or default directory. Before this function user must set DTA (Disk transfer area) using function 1AH.	If function succeeds, CY = 0, i.e. matching filename is found and the following information is reported in DTA- Bytes 00-14H-reserved 15H-Attribute of the file 16H-17H-Time of creation/update

(Contd.)

Function value in AX with AH	Function	Register I/P	Return O/P
			18H-19H-Date of creation/update 1A-1BH-Least significant word of file size 1C-1DH-Most significant word of file size 1E-2AH-Filename and extension of the matched file if function fails, CY = 1 i.e. a matching filename is not found and AX = 02H-Invalid path 12H-If no file with the matching name is found in the default or specified directory.
80. Function 4FH	Search the default or specified directory for next matching file, following a successful call to function 4EH, and report various information about it.	AH = 4FH, Function 1 AH must be called to set DTA before executing this function.	If function succeeds, CY = 0 i.e. a matching filename is found and the following information is reported in DTA- Bytes 00-14H-reserved 15H-Attribute of the file 16H-17H-Time of creation/update 18H-19H-Date of creation/updated 1A-1BH-Least significant word of file size 1C-1DH-Most

(Contd.)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
			significant word of file size 1E-2AH-Filename and extension of the matched file If function fails, CY = 1 i.e. a matching filename is not found and AX = 12 H-If no file with the matching name is found in the default or specified directory
81. Function 50H-53H	RESERVED		
82. Function 54H	Get verified state-Using this function every disk write operation can be verified for correctness of the written data by reading it after the write operation.	AH = 54 H	AL = 00-If varify flag is off 01-If varify flag is on
83. Function 55H	RESERVED		
84. Function 56H	Rename the existing file contains Segment: Offset string of the file to be contains Segment: Offset	AH = 56H, DS:DX and the file is of the null terminated CY = 1 and-AX = 02- renamed and ES:DI 03-Invalid Path of the null terminated string of the new filename.	If successful,CY = 0 renamed, else File not found 05-Access denied 11H-for not the same device.
85. Function 57H	Handle type call to get or set the date and time stamp of a previously opened file new time, DX = 16-bit	AH = 57H, BX = file handle of the previously opened file, AL = 00 if getting date and time AL = 01 if setting date and time. If AL = 01, CX = 16-bit is invalid new date information	If function fails, CY= 1 and AX = 01H-If invalid option is and indicated in AL for calling function. 06H-If handle in BX If function successes, CY = 0 For AL = 00, on return CX = 16-bit time stamp of the file DX = 16-bit date

(Contd.)

Function Value in AX with AL	Function	Register I/P	Return O/P
			stamp of the file For AL = 01, on return time and date fields of the file are modified appropriately.
86. Function 58H	Get or set memory allocation strategy If getting strategy, AL	AH = 58H, I and AX = 1 in = 00. If setting strategy, AL = 01	If function fails, CY = dicating invalid option exercised through AL on call. If function succeeds,
		CY = 0 and (a) if	strategy was being set nothing is returned. (b) if strategy is being read i.e. AL = 00 while calling, AX = current strategy
		code.	
87. Function 59H	Extended error reporting function	AH = 59H, BX = 00H code	AX = Extended error
			BH = Error class BL = Recommended
		action for the	reported error CH = Error locus i.e.
		device where	error occurrence.
88. Function 5AH	Create temporary file of temp file. 00-normal,	AH = 5AH, CX = Attribute Path not found, AX = 01-read-only 02-hidden, 04-system, DS:DX points to null terminated	If fails CY = 1, AX = 5-Access denied. If succeeds CY = 0, AX = file handle of new file,
	string format	filename path in ASCII temporary filename	DS:DX points to the
path.			
89. Function 5BH	Create new file Attribute of temp file. 00-	AH = 5AH, CX = 03-Path not found,	If fails CY = 1, AX =

(*Contd.*)

Function Value in AX/AH AL	Function	Register I/P	Return O/P
	02-hidden, 04-system, in ASCII string format.	normal, 01-read-only available handle, AX DS:DX points to null terminated filename path created already	AX = 06 - No - 05- Access denied, AX = 50H-File to be exists. If succeeds CY = 0, AX = file handle of new file
90. Function 5CH-5FH	These are intended for use of networking and are not of interest as far as this text is concerned.		
91. Function 60H-61H	RESERVED		
92. Function 62H	Get the address of the current program PSP	AH = 62H	BX = Segment address of the current program PSP
93. Function 63H	This function was used in DOS 2.25 only and is not of interest here.		

N.B.

1. There are number of other DOS interrupt functions. This Appendix enlists only the functions under INT21H in brief. For details of these functions users may refer 'Microsoft DOS Encyclopedia or 'Microsoft DOS Reference Manual'.
2. Besides the DOS interrupts the personal computers also provide a separate family of BIOS interrupts. Their details may be obtained from 'IBMPC Reference Manual'.