

5 INPUT/OUTPUT ALLOCATION (TRAP #2)

Input and output QDOS procedures fall into two major categories. First, there are those that are concerned with the actual allocation of channels for devices and files. Second, there are those that actually perform input and output operations on the allocated channels. The first category comprises the TRAP #2 calls discussed in this chapter. The second category comprises the TRAP #3 calls discussed in Chapter 6.

5.1 Redirectable input and output

An important concept within QDOS is that of redirectable I/O. It is not necessary for an applications programmer to know what physical device is actually attached to an I/O channel. Opening a channel for use with a Microdrive involves accessing exactly the same routine as when opening an extra screen window channel. The real task of getting information in and out of channels is the job of the device driver.

Sometimes it is necessary for certain aspects of physical devices to be specified alongside the actual device name. This additional information is used by the appropriate device driver to configure the channel as necessary.

5.2 Standard device names

QDOS supports a range of standard device drivers. Each driver can accept a pre-defined name followed by any appropriate configuration parameters. Known devices, and their parameters, are as follows:

`CON_wXhAxYy_k` Console I/O. A window is defined as width 'w' by height 'h', at location 'x', 'y'. The keyboard type-ahead buffer is set to length 'k'. Note that the size and position of the window is defined in terms of pixels on a 512 by 256 pixel grid. The width 'w' and the 'x' coordinate should both be specified in multiples of two pixels.
The default definition is: `CON_448x180a32x16_128`

SCR_wXhAxYy Screen output. This has the same window definition as for the I/O device 'CON'.
The default definition is: SCR_448x180a32x16

SERnpz Serial I/O. The port number is specified by 'n'. Parity (Even, Odd, Mark, or Space) is defined by 'p' being one of E, O, M, or S. Parameter 'z' specifies protocol by taking on one of the values R (raw data), Z (CTRL Z is eof), or C (<CR> is changed to <LF> on input, <LF> is changed to <CR> on output, and CTRL Z is eof).
The default definition is: SER1R

NETI_nn Serial network link from node 'nn'.

NETO_nn Serial network link to node 'nn'.

MDVn_name Microdrive. Microdrive number 'n' and file name given by 'name'. There is no default.

PIPE_n
PIPE

5.3 Use of 68000 registers

The TRAP #2 procedures are accessed with register D0 (byte) indicating which particular call is required. This register is used also to return an error status (long-word) to the calling process. If the error code returned is not zero then an error has occurred. Small negative error codes are used to indicate standard errors. These error codes are listed in Appendix C. If the trap call invoked some form of additional device driver, the error code returned can be a pointer to a specific error message. In order that the two types of error return code might never be confused, the pointer type error code is in fact a pointer to an address \$8000 below that of the true error message. Potentially, all QDOS routines can return the error 'ERR.BP' (-15), signifying 'bad parameter'. The full descriptions of the TRAP #2 procedures state which additional errors can be returned. It would of course be wise to check for any errors after the trap call has been made.

In addition to the use of register D0, data registers D1 to D3 and address registers A0 to A3 are variably used to pass values to and from the QDOS procedures. When the appropriate registers have been set for any one call the appropriate routine is accessed by simply executing the TRAP #2 instruction. In cases where the data size qualifier (i.e., '.B', '.W', or '.L') is not specified within the description, the default is long-word (i.e., '.L').

IO.OPEN \$01 (1)

Open a channel

Entry parameters:	D1.L	Job ID
	D3.L	Access code
	AO	Address of channel name
Return parameters:	D1	Job ID
	AO	Channel ID
Affected registers:	D1, AO	
Additional errors:	NJ (-2)	not a valid job
	OM (-3)	out of memory
	NO (-6)	too many channels open
	NF (-7)	file or device not found
	EX (-8)	file already exists
	IU (-9)	file or device in use
	BN (-12)	bad device or file name

Description

A channel is opened by specifying the appropriate device and/or file name. On entry, AO is a pointer to the name, which is a string of ASCII characters preceded by a character count (data length .W), this being the standard format for a string parameter. Register AO points to the word of data representing the character count.

Each job requires its own list of I/O channels and therefore the job ID must be supplied on entry to the procedure. If the job ID is passed as a negative integer (e.g., -1), the channel will be associated with the current job.

More than one type of open is available, and on entry D3 may hold a code specifying which open option is required. The options available, together with their access codes, are as follows:

Access code	Type
0	Opens existing file/device. Exclusive use. R/W.
1	Opens existing file/device. Shared use. R/O.
2	Opens new file. Exclusive use. R/W.
3	Opens new file (overwrite).
4	Opens directory file. R/O.

Most device drivers ignore the access code. QDOS does support the 'overwrite' type of access, but the Microdrive device driver does not.

The error ERR.BN (-12) is usually given when the name of the device has been recognized, but the parameter information is in an incorrect form. A channel will not be opened if any error occurs.

IO.CLOSE \$02 (2)

Close a channel

Entry parameters: AO Channel ID
Return parameters: none
Affected registers: AO
Additional errors: NO (-6) channel not open

Description

This procedure simply closes the specified channel.

IO.FORMAT \$03 (3)

Format a sectorized medium

Entry parameters:	A0	Pointer to medium name
Return parameters:	D1.W	Number of good sectors
	D2.W	Total number of sectors
Affected registers:	D1, D2, A0	
Additional errors:	OM (-3)	out of memory
	NF (-7)	drive not found
	IU (-9)	drive in use
	FF (-14)	format failed

Description

The medium name pointed to by A0 on entry to the procedure must be in the standard string parameter form. This means that A0 will be pointing to a word of data (specifying the length of the string) followed by the ASCII string itself.

The string should consist of the drive name, followed by the drive number, then an underscore, and finally the medium name (up to 10 characters). The medium name is that name which will be supplied when a directory is performed on the medium. For example, the command string:

MDV1_MYPROGS

will format the medium in Microdrive-1, and label the medium 'MYPROGS'.

IO.DELET \$04 (4)

Delete a file

Entry parameters: D1.L Job ID
A0 Address of device/file name

Return parameters: none

Affected registers: D1, D3, A0 - A2

Additional errors: OM (-3) out of memory
NO (-6) too many channels open
NF (-7) file or device not found
BN (-12) bad device or file name

Description

Deleting a file is a form of 'open' operation and therefore a job ID must be supplied. Any job ID may be used; -1 being the most convenient (i.e., the current job). On entry, A0 is a pointer to the name, which is a string of ASCII characters preceded by a character count (data length .W), this being the standard format for a string parameter. Register A0 points to the word of data representing the character count.

No delete operation will have been performed if any error has occurred.