2 OPERATING SYSTEM CALLS

The list below contains all the Acorn supported operating system routines and their vectors which exist in the Electron OS 1.0. See the User Guide for a general description of these calls.

2.1 OSWRCH Write character routine

Call address &FFEE Indirected through &20E

This routine outputs the character in the accumulator to the currently selected output stream(s).

On exit:
    A, X and Y are preserved.
    C, N, V and Z are undefined.

    The interrupt status is preserved (though interrupts may be enabled during a call).

2.2 Non-vectored OSWRCH

Call address &FFCB

This call is normally made by OSWRCH. This call has no vector and so cannot be intercepted. Its use is not recommended for this reason.

2.3 OSRDCH Read character routine

Call address &FFEO Indirected through &210

This routine reads a character from the currently selected input stream and returns it in the accumulator.

On exit:
C=0 indicates that a valid character has been read.
C= 1 indicates that a character has not been read due to an error.

f an error should occur acknowledgement of the error condition should be made using OSBYTE &7E.

X and Y are preserved.
N, V and Z are undefined.

The interrupt status is preserved (though interrupts may be enabled during a call).

2.4 Non-vectored OSRDCH

Call address &FFC8

This call is normally made by OSRDCH, it is not available for interception and its use is not recommended by Acorn.

2.5 OSNEWL Write a newline routine

Call address &FFE7 Not indirected

This routine writes a line feed (&A/10) and a carriage return (&D/13) to the current output stream(s) using OSWRCH.

On exit:
    A=&0D (13)
    X and Y are preserved.
    C, N, V and Z are undefined.

    Interrupt status is preserved (though it may be enabled during a call).

2.6 OSASCI Write character routine, OSNEWL called if A=&0D (13).

Call address &FFE3 Not indirected

This is a write character routine performing the same action as OSWRCH but which outputs a line feed and a carriage return in response to a carriage return character.

On exit:
    A, X and Y are preseved.
    C, N, V and Z are undefined.

    Interrupt status is preserved (though interrupts may be enabled during a call).

2.7 GSINIT General string input initialise routine.

Call address &FFC2

The original intention was that this routine together with GSREAD would provide a standard string input facility for the use of filing system paged ROMs. It is now felt that this routine is unsuitable for that purpose and accordingly its use is not recommended.

This routine initialises a string for input prior to reading using GSREAD.

Entry parameters:
    String address stored in &F2 and &F3 plus offset in Y
    C=0, if first space, CR or second " terminates input
    C=1, if first space does not terminate input

On exit:
    Y contains the offset of the first non-blank character from the address contained in &F2 and &F3.

    A contains the first non-blank character of string

    Z flag is set if the string is a null string

2.8 GSREAD Read character from string input routine.

Call address &FFC5

This routine is used to read characters from an input string after a GSINIT call. Control codes and non-ASCII values may be introduced into the input string by using an escape character, '|'. The escape character followed by a letter gives a character value equal to the ASCII value minus 64 (&40). The escape character followed by a '!' character gives a value of 128 plus the value of the next character in the string. An escape character followed by

itself gives the escape character.

Entry parameters:
&F2, &F3 and Y set by GSLNLT

C=0 String terminated by first space, carriage return
or second quotation mark.
C= 1 String terminated by carriage return
or second quotation mark.

On exit:
A contains the character read from the string.
Y contains the index for the next character to be read.
C=1 if the end of string is reached.
X is preserved.

2.9 OSRDRM Read byte from paged ROM routine.

Call address &FFB9

Entry parameters:
ROM number stored in Y.
Address stored in &F6 and &F7.

This call returns a byte read from a paged ROM.

On exit:
A contains the value of the byte read.

This routine was included for the implementation of ROM filing
system software in paged ROM and is not recommended for
general use.

2.10 OSEVEN Generate an event routine.

Call address &FFBF

The user event may be generated using this routine. Software replacing OS routines should generate the appropriate events by making this call.

Entry parameter:
        The event number should be placed in Y.

On exit:
        C=0 if and only if the event was enabled.

2.11 OSCLI Pass string to the CLI.

Call address &FFF7 Indirected through &208

This routine is implemented on the BBC micro, the Electron and the Tube operating system.

This call provides the machine code user with a convenient method of performing any of the ***commands** that the system provides from Basic. The command required is placed in a string as normal text and this call is made.

If the string passed to the CLI is not terminated by a carriage return within 255 bytes this routine has undefined effects.

The following commands are recognised:

*|	*String escape character* rest of command ignored
*.	treated as a *CAT command
*/	treated as a *RUN command
*BASIC	select BASIC as current language
*CAT	issue catalogue request to filing system
*CODE	passed to user vector (see chapter 6)
*EXEC	select text file as input stream

*FX	issue OSBYTE call (no registers returned)
*HELP	issue paged ROM service call 9, see chapter 10
*KEY	take rest of line as text for soft key
*LINE	passed to user vector (see chapter 6)
*LOAD	issue load request to filing system
*MOTOR	open/close cassette motor relay
*OPT	issue option request to filing system
*ROM	select *ROM filing system
*RUN	issue load and execute request to filing system
*SAVE	issue save request to filing system
*SPOOL	include text file in output stream
*TAPE	select tape filing system
*TV	ignored by the Electron

These commands may be abreviated by taking the first few letters and terminating with a '.' character. Parameters may be passed in the text following the command.

Other *unrecognised commands* are first offered to paged ROMs (see section 10.1) and are then offered to the currently selected filing system via the filing system control vector (see chapter 5).

Entry parameters:
X and Y contain the address of a line of text
(X=low-byte, Y=high-byte) terminated by a CR character.

On exit:
A, X, Y, C, N, V and Z are undefined.
Interrupt status is preserved but interrupts may be enabled during a call.