

NAME

getc, getw, fopen – buffered input

SYNOPSIS

```

mov    $filename,r0
jsr    r5,fopen; iobuf

fopen(filename, iobuf)
char *filename;
struct buf *iobuf;

jsr    r5,getc; iobuf
(character in r0)

getc(iobuf)
struct buf *iobuf;

jsr    r5,getw; iobuf
(word in r0)

getw(iobuf)
struct buf *iobuf;

```

DESCRIPTION

These routines provide a buffered input facility. *Iobuf* is the address of a 518(10) byte buffer area whose contents are maintained by these routines. Its structure is

```

struct buf {
    int fildes;      /* File descriptor */
    int nleft;      /* Chars left in buffer */
    char *nextp;  /* Ptr to next character */
    char buff[512]; /* The buffer */
};

```

Fopen may be called initially to open the file. On return, the error bit (c-bit) is set if the open failed. If *fopen* is never called, *get* will read from the standard input file. From C, the value is negative if the open failed.

Getc returns the next byte from the file in r0. The error bit is set on end of file or a read error. From C, the character is returned as an integer, without sign extension; it is -1 on end-of-file or error.

Getw returns the next word in r0. *Getc* and *getw* may be used alternately; there are no odd/even problems. *Getw* may be called from C; -1 is returned on end-of-file or error, but of course is also a legitimate value.

Iobuf must be provided by the user; it must be on a word boundary.

To reuse the same buffer for another file, it is sufficient to close the original file and call *fopen* again.

SEE ALSO

open (II), read (II), getchar (III), putc (III)

DIAGNOSTICS

c-bit set on EOF or error; from C, negative return indicates error or EOF. Moreover, *errno* is set by this routine just as it is for a system call (see introduction (II)).

BUGS