

## NAME

getc – 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 format is:

```

ioptr:  .=.+2           / file descriptor
          .=.+2           / characters left in buffer
          .=.+2           / ptr to next character
          .=.+512. / the buffer

```

Or in C,

```

struct buf {
    int fildes;
    int nleft;
    char *nextp;
    char buffer[512];
};

```

*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; 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* is 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), putc(III)

## DIAGNOSTICS

c-bit set on EOF or error;  
from C, negative return indicates error or EOF.

GETC (III)

4/30/72

GETC (III)

**BUGS**