NAME

>   putc, putw, fcreat, fflush − buffered output

SYNOPSIS

>   **mov $filename,r0**
>   **jsr   r5,fcreat; iobuf**
>
>   **fcreat(file, iobuf)**
>   **char *file;**
>   **struct buf *iobuf;**
>
>   (get byte in r0)
>   **jsr   r5,putc; iobuf**
>
>   **putc(c, iobuf)**
>   **int c;**
>   **struct buf *iobuf;**
>
>   (get word in r0)
>   **jsr   r5,putw; iobuf**
>
>   **putw(w, iobuf);**
>   **int w;**
>   **struct buf *iobuf;**
>
>   **jsr   r5,flush; iobuf**
>
>   **fflush(iobuf)**
>   **struct buf *iobuf;**

DESCRIPTION

>   *Fcreat* creates the given file (mode 666) and sets up the buffer *iobuf* (size 518 bytes); *putc* and
>   *putw* write a byte or word respectively onto the file; *flush* forces the contents of the buffer to be
>   written, but does not close the file. The structure of the buffer is:
>   **struct buf {**
>   **int fildes;   /* File descriptor */**
>   **int nunused;   /* Remaining slots */**
>   **char *xfree;   /* Ptr to next free slot */**
>   **char buff[512]; /* The buffer */**
>   **};**
>
>   Before terminating, a program should call *flush* to force out the last of the output (*fflush* from C).
>
>   The user must supply *iobuf,* which should begin on a word boundary.
>
>   To write a new file using the same buffer, it suffices to call *[f]flush,* close the file, and call *fcreat*
>   again.

SEE ALSO

>   creat (II), write (II), getc (III)

DIAGNOSTICS

>   *Fcreat* sets the error bit (c-bit) if the file creation failed (from C, returns −1). *Putc* and *putw* re-
>   turn their character (word) argument. In all calls *errno* is set appropriately to 0 or to a system er-
>   ror number. See introduction (II).

BUGS