

NAME

exit - terminate process

SYNOPSIS

```
exit (status)
struct { char lobyte; char hbyte; } status;
_exit(status)
```

DESCRIPTION

Exit is the normal means of terminating a process. *Exit* closes all the process' files and notifies the parent process if it is executing a *wait*. The low byte of r0, *status.lobyte*, is available as status to the parent process via *wait*.

There are two C callable versions. *Exit* calls the user definable routine `_cleanup` to perform any user defined cleanup actions; then it does an *exit*. The C library version of `_cleanup` - which is used if the user does not supply his own - simply returns. The other version, `_exit`, exits without calling anything. It is provided so that users may write their own version of *exit*.

When a process dies, e.g. via *exit*, the child death signal, **SIGCLD**, is sent to its parent (see *signal(2)*).

This call can never return.

SEE ALSO

wait(2), *signal(2)*, *fclose(3S)*

ASSEMBLER

```
(exit = 1.)
(status in r0)
sys exit
```