

The "ilock" Module

June 23, 1967

Art Evans

The file system people have written a complete set of interlocking routines. BG.15.02 will eventually be updated (it is currently quite out of date), but in the mean time the following discussion may be of help to others. In all cases, the first parameter is a pointer to the location to be interlocked and the other parameter(s) are fixed. The interlock value used will be the value in pds\$process_id.

The statement

```
call ilock$trylock(ptr, to_me, to_someone_else);
```

will first set the second and third parameters to zero and then try a stac into the location pointed to. If it succeeds, return will be immediate. If not, the identification of the current process will be compared with that in the lock. If they are the same, the second parameter will be set to one, while otherwise the third parameter will be set to one.

The statement

```
call ilock$looplock(ptr, already_locked);
```

will wait in a loop until the interlock has been locked by the calling process, and will return with the second parameter set to zero. If the interlock was already set by the calling process, return will be immediate with the second parameter set to one.

The statement

```
call ilock$loopunlock(ptr);
```

will unlock the interlock. If it was not previously locked to the calling process, a panic stop to bring down the system will be made.

These routines are all in the system library.