3.1.9 T_MODE

NAME

t_mode -- "Change Task Mode"

SYNOPSIS

```
uint t_mode ( mode, mask, &pmode )

        uint mode;     /* new mode */
        uint mask;     /* mask */
        uint pmode;    /* previous mode - returned by this call */
```

The *mode* and *mask* values are defined as follows:

NOPREEMPT	set	to disable preempting
	clear	to enable preempting
TSLICE	set	to enable timeslicing
	clear	to disable timeslicing
NOASR	set	to disable asynchronous signal processing
	clear	to enable asynchronous signal processing
SUPV	set	to execute in supervisor mode
	clear	to execute in user mode
LEVEL		interrupt level when SUPV is set

DESCRIPTION

T_mode enables and disables several modes of execution for the calling task. A task may enable/disable timeslicing, enable/disable preempting, enable/disable asynchronous signal processing, or execute in supervisor mode at an optional interrupt level.

Tasks have the ability to process signals asynchronously. Any task with a valid asynchronous signal routine (asr) which needs to temporarily disable asynchronous processing should use this directive.

To change a particular mode, the user must indicate which mode is being changed by setting the appropriate value in the *mask* parameter, and then set the appropriate value in the *mode* parameter to the new mode. For example, if the user only wants to change the preempt mode characteristic, he would set the mask value to NOPREEMPT and the mode value to NOPREEMPT to disable preempting, or the mode field to 0 to enable preempting.

If the preempt mode is not in effect, timeslicing will not take place.

RETURN VALUE

The *t_mode* call always succeeds, *pmode* is filled in, and 0 is returned.

NOTES

Not callable from ISR.

May cause a preempt if the running task enables preempting.

Refer to *as_catch* for discussion on receiving asynchronous signals.

3.1.10 T_GETREG

NAME

t_getreg – "Get a task's register"

SYNOPSIS

uint t_getreg (tid, regnum, &regval)

```
        uint tid;         /* task id as returned from t_create or t_ident */
        uint regnum;      /* register number */
        uint regval;      /* register value - returned by this call */
```

The *regnum* field values are:

S_REG0	System defined register 0
S_REG1	System defined register 1
S_REG2	System defined register 2
S_REG3	System defined register 3
S_REG4	System defined register 4
S_REG5	System defined register 5
S_REG6	System defined register 6
S_REG7	System defined register 7
U_REG0	User defined register 0
U_REG1	User defined register 1
U_REG2	User defined register 2
U_REG3	User defined register 3
U_REG4	User defined register 4
U_REG5	User defined register 5
U_REG6	User defined register 6
U_REG7	User defined register 7

DESCRIPTION

The executive returns the register value in the *regval* field for the register identified in the *regnum* field and the task identified by the *tid*.

The task identified in the *tid* field may exist on the local processor, or any remote processor in the multiprocessing configuration if the task was created with the GLOBAL flags value set (see *t_create)*.

RETURN VALUE

If *t_getreg* is successful, *regval* is filled in, and 0 is returned.

If the call was not successful, an error code is returned.

ERROR CONDITIONS

Invalid *tid*.

Invalid register number.

ISR cannot reference remote node.

NOTES

Can be called from within an ISR, except when the task was not created on the local node.

Will not cause a preempt.

3.1.11 T_SETREG

NAME

t_setreg -- "Set a task's register"

SYNOPSIS

```
uint t_setreg ( tid, regnum, regval )

        uint tid;       /* task id as returned from t_create or t_ident */
        uint regnum;    /* register number */
        uint regval;    /* register value  */
```

The *regnum* field values are:

S_REG0	System defined register 0
S_REG1	System defined register 1
S_REG2	System defined register 2
S_REG3	System defined register 3
S_REG4	System defined register 4
S_REG5	System defined register 5
S_REG6	System defined register 6
S_REG7	System defined register 7
U_REG0	User defined register 0
U_REG1	User defined register 1
U_REG2	User defined register 2
U_REG3	User defined register 3
U_REG4	User defined register 4
U_REG5	User defined register 5
U_REG6	User defined register 6
U_REG7	User defined register 7

DESCRIPTION

The executive sets the register identified in the *regnum* field for the task identified by the *tid* with the value in the *regval* field.

The task identified in the *tid* field may exist on the local processor, or any remote processor in the multiprocessing configuration if the task was created with the GLOBAL flags value set (see *t_create)*.

RETURN VALUE

If *t_setreg* successfully set the register value, 0 is returned.

If the call was not successful, an error code is returned.