

7. TASK SCHEDULING AND ACTIVATION

7.1 Dispatcher queue

It has been previously stated that, in a system when more than one task exists, the tasks are 'considered for dispatching' when a LKM is performed. The mechanism for task dispatching and queuing, of tasks will now be outlined.

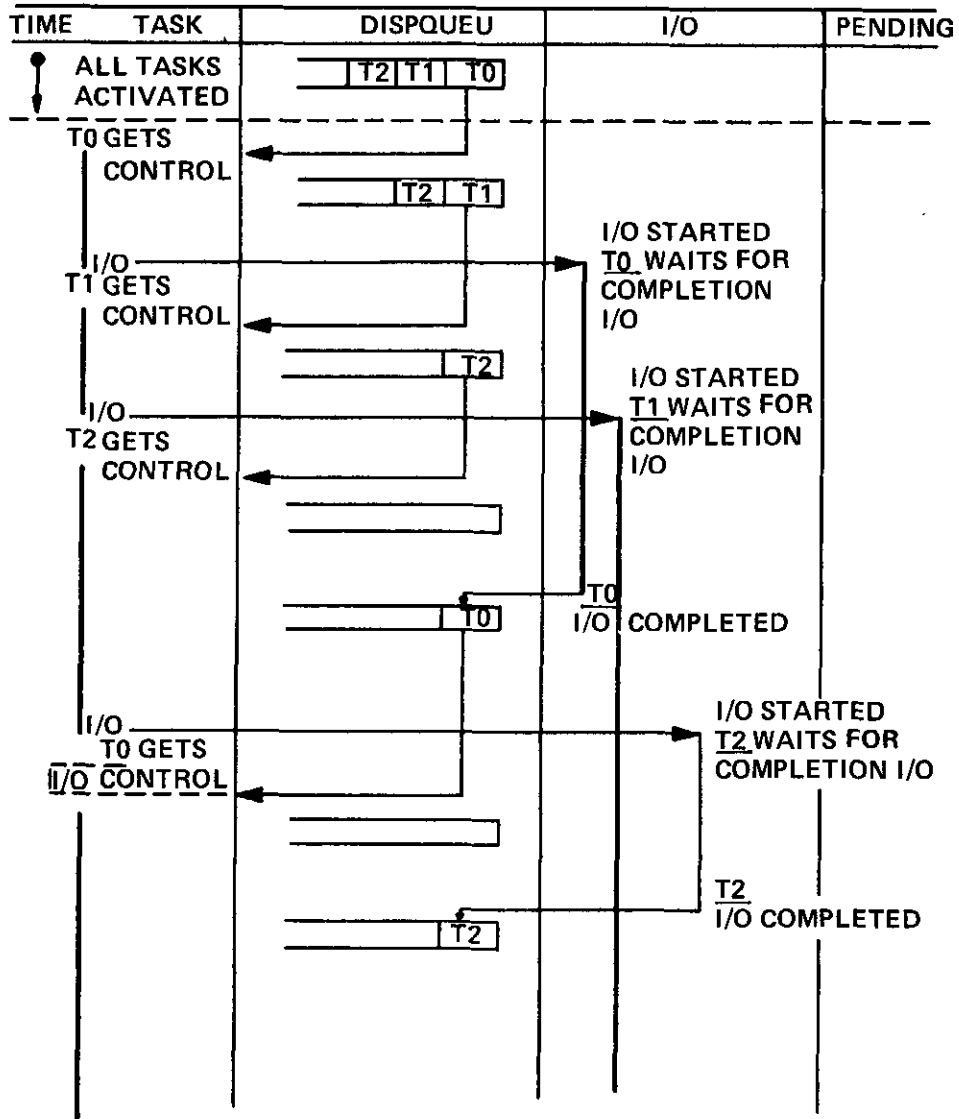
The scheduling of tasks is performed by the TOSS Monitor: at system start, all the tasks are placed in the Dispatcher queue, and the first task is then activated, i.e. starts execution.

A task runs in User mode, which means that when an I/O instruction is executed, control passes to the Monitor. The Monitor then starts the I/O operation. The task can not continue until the I/O is complete, assuming Wait is used, and this means that the task is put at the back of the dispatcher queue, which operates on a FIFO principle, and another task may get control. This principle is necessary particularly for keyboard input, which is relatively slow, since otherwise the keyboard input task would hold up the other tasks in the system. It is quite possible that the dispatcher queue can be empty, if all tasks are waiting for I/O to be completed. In this case the Monitor is in an 'idle loop', until such time as I/O completes for one of the tasks. There are other ways in which the dispatcher queue can be affected and these are the instructions:

- PAUSE inhibits execution of the task, until restarted by another task; during this time the task is pending, and is not considered for dispatching.
- RSTRT restarts the specified task; the task does not necessarily get control, but is placed at the back of the dispatcher queue.
- EXIT terminates the task, and it ceases to exist; task tables and all references to the task are deleted.
- ACTV activates a task, which may or may not have already been active and performed an EXIT. Activation may be at any statement that contains a label. Again, the task may not get control, but is placed in the dispatcher queue.

Keyword	Page in manual
PAUSE	MO4 1.4.133
RSTRT	MO4 1.4.142
EXIT	MO4 1.4.106
ACTV	MO4 1.4.20

SCHEDULING OF TASKS



7.2 More than one start point

It is possible to specify more than one start point in a module. At system start, the task will be activated at both start points. However, since the task can not exist twice, one start point is placed in the dispatcher queue, and the other in the pending queue.

When the task EXITS, the task is deactivated, and the second start point is removed from the pending queue and placed in the dispatcher queue, and is thereby activated at the second start point.

MULTIPLE START POINTS

START PNTS → S1,S2

