

Date: 15 April, 1986

Subject: Scheduler

Author: John Worthington

Document Version Number: 00.20

Revision History

00.00	(3 Feb 86)	Initial Release
00.10	(12 Feb 86)	Vectors for BusyFlag added.
00.20	(15 Apr 86)	Modified to match Beta firmware

Overview:

One of the problems with interrupt based desk accessories is that a large portion of the code in the system is not reentrant. Activating a desk accessory within a non-reentrant piece of code will almost always generate a system failure. As a solution to this problem, this document introduces a busy flag for non-reentrant serially reusable resources and the Scheduler -- a mechanism for delaying the activation of desk accessories and other tasks that require the use of a currently busy resources.

It should be noted that Scheduler provides only the minimal functionality necessary to support desk accessories, etc. Scheduler should not be used to provide a pseudo-multitasking environment.

Using the Scheduler:

Scheduler revolves around a busy flag located at \$E1/00FF. The flag is referred to as BUSYFLAG. Code wishing to change the state of the busy flag should use the routines INCBUSYFLG and DECBUSYFLG.

Both INCBUSYFLG and DECBUSYFLG are accessed from vectors in bank \$E1:

INCBUSYFLG	\$E1/0064
DECBUSYFLG	\$E1/0068

When a piece of non-reentrant code is entered, it should do a JSL to INCBUSYFLG in full native mode (e=0, m=0, x=0). When exiting from the piece of code a JSL to DECBUSYFLG should be performed. DECBUSYFLG will decrement the busy flag and execute any tasks that might have been placed in Scheduler's que. All registers are preserved across calls to INCBUSYFLG and DECBUSYFLG.

Adding to Scheduler's Que:

Scheduler provides a que with space for 4 items. This should be sufficient for supporting the Desk Manager, as well as other small interrupt handlers.

I considered adding room for more tasks to the Scheduler's que, but decided that this might decrease the focus of Scheduler. Scheduler is not intended to provide multitasking.

The tool call SCHADDTASK is used to add a task to Scheduler's que.

input	(space for result)	word
->	pointer to task	long
output	result	

The result will be non-zero if the task was added to the que and zero otherwise.

When the busy flag is decremented to zero, the tasks in the que are executed in the order they were posted. The routines will be called from a JSL in full native mode with the data bank register set to \$E0.

It is also possible to flush Scheduler's task que. This is accomplished with the tool call SCHFLUSH. It requires no parameters. Applications should never make this call.

Scheduler Calls

Tool Number 7

Calls	Number	Effect
SCHBOOTINIT	01	Initializes flags and counters.
SCHAPPINIT	02	nothing
SCHAPPTERM	03	nothing
SCHVERSION	04	Returns version information.
SCHRESET	05	Reinitializes flags and counters.
SCHACTIVE	06	Returns true if Scheduler is active.
(reserved)	07	
(reserved)	08	
SCHADDTASK	09	Adds task to Scheduler's que.
SCHFLUSH	0A	Flushs Scheduler's que.