

REAL TIME OPERATING SYSTEM PROGRAMMING-I: μ C/OS-II and VxWorks

Lesson-4: μ C/OS-II Time set, get and delay functions

1. System Time and Time Delay Functions

OSTimeSet (*counts*)

- void OSTimeSet (unsigned int *counts*)

Used when system time is to be set by *counts* (Example 9.10 Step 10)

OSTimeGet() and OSTimeDly (delayCount)

- unsigned int_OSTimeGet (void)

To find present *counts* when system time is read. (Example 9.11 Step 1)

- void OSTimeDly (unsigned short delayCount)

To delay a task by period of count-inputs equal to delayCount -1 (Example 9.12 Step 20)

OSTimeDlyHMSM (hr, mn, sec, ms)

- void OSTimeDlyHMSM (unsigned byte hr, unsigned byte mn, unsigned byte sec, unsigned short ms)

When need is to delay and block a task for hr hours, mn minutes, sec seconds and ms milliseconds

- OSTimeDlyHMSM (0, 0, 0, 1000)
at Example 9.12 step 19

OSTimeDlyResume (taskPriority)

- unsigned byte_OSTimeDlyResume (unsigned byte_taskPriority)

When a task of priority = taskPriority is to resume before the preset delay, which was by a value defined either by delayCount or (hr, mn and ms) and which is in blocked state now (Example 9.12 Step 19)

Macros to find status after execution of OS Time Delay Functions

- OS_NO_ERR, when our arguments are valid and resumption after delay succeeds.
- OS_TIME_INVALID_HOURS,
- OS_TIME_INVALID_MINUTES,
- OS_TIME_INVALID_SECONDS and
- OS_TIME_INVALID_MILLI, returns true if the arguments are greater than 55, 59, 59 and 999, respectively.

Macros to find status after execution of OS Time Delay Functions

- `OS_TIME_ZERO_DLY`, returns true if all the arguments passed are 0.
- `OS_NO_ERR` returns true if all the arguments passed or when resumption after delay succeeds.
- `OS_TASK_NOT_EXIST` returns true, if task was not created earlier.

Macros to find status after execution of OS Time Delay Functions

- OS_TIME_NOT_DLY returns true, if the task was not delayed.
- (vi) OS_PRIO_INVALID returns true, when taskPriority parameter that was passed is more than the OS_PRIO_LOWEST (23) when maximum number of user tasks = 8

Summary

We learnt

- **μC/OS-II** Functions for initial setting of system-time, starting system ticks, initiating OS
- **μC/OS-II** Task delay and delay-resume functions

End of Lesson-4 Chapter 9 on
 μ C/OS-II Time set, get and delay
functions