

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

Lesson-8: **μ C/OS-II Queue IPC functions**

1. Queue Functions

Queue Functions

- The message pointers post into a queue by the tasks either at the back as in a *queue* or at the front as in a *stack*.
- A task can thus insert a given message for deleting either in the *first in first out* (FIFO) mode or in *priority* mode for priority message.

Queue Functions

- Assume pointer, `**Qtop`, to a queue of pointers for the messages and
- Assume two pointers, `*QfrontPointer` and `*QbackPointer` to insert (post) and delete (retrieve), respectively, the pointer of the message.

OSQCreate (**QTop, qSize)

- OS_Event OSQCreate (void **QTop, unsigned byte qSize)

/*OS creates a Queue ECB. This creates and initialises an array of pointers for the queue at QTop*/ .

OSQCreate (**QTop, qSize)

- Queue can be of maximum size = *qSize*.
- QTop points to top (0th element of an array). ECB points at the QMsgPointer.
(Example 11.20 Step 8)

OSQPost (*QMsgPointer, *QMsg)

- unsigned byte OSQPost
(OS_EVENT*QMsgPointer, void
*QMsg)

—Sends a pointer of the message QMsg to the QMsgPointer at the queue back. The message inserts at the queue back pointer in the ECB (Example 11.20 step 21)

OSQPostFront (*QMsgPointer, *QMsg)

- unsigned byte OSQPostFront
(OS_EVENT *QMsgPointer, void *QMsg)

—Sends QMsg to the QMsgPointer at the queue. It points to the queue front pointer in the ECB where pointer for QMsg now stores pushing other message-pointers back. (Example 11.20 step 20)

OSQPend (*QMsgPointer, timeOut, *Qerr)

- void *OSQPend (OS_Event *QMsgPointer, unsigned short timeOut, unsigned byte *Qerr)
 - The message pointer points to the queue front (head) at the ECB for the queue defined by QMsgPointer. It suspends the task *QMsgPointer is Null (until either message inserts or timeout ticks of RTOS timer). (Example 11.20 Step 39)

OSQFlush(*QMsgPointer)

unsigned byte *OSQFlush
(OS_EVENT *QMsgPointer)

— To eliminate all the messages in the queue that have been sent. This function checks if a queue has a message pending at QMsgPointer.

OSQQuery (*QMsgPointer, *QData)

- unsigned byte OSQQuery
(OS_EVENT *QMsgPointer,
OS_Q_DATA *QData)

—To get queue message information and error information.

2. Macros for queue functions to find status after execution of OS queue Functions

Macros for queue functions

OS_NO_ERR,

returns true when querying or insertion or deleting succeeds

OS_ERR_EVENT_TYPE,

if *QMsgPointer is not pointing to queue message.

Summary

We learnt

- μ C/OS-II s queue functions.
- A queue can be posted (inserted) from the sender tasks an array of message pointers.
- Message pointer insertions can be such that later on it can retrieve in FIFO (first-in first-out) mode as well as priority mode from a queue

End of Lesson-8 on
 μ C/OS-II Queue IPC functions