

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

Lesson-12: **VxWorks functions for Queue**

1. Queue functions

Queue Functions

- VxWorks queue functions at library, msgQLib, which the user includes for using these.
- For full duplex communication between two tasks, create two queues, one for each task.

Queue Create and Delete Functions

- `msgQCreate ()`
–to allocate and initialize a queue (Example 9.24 Step 8)
- `msgQDelete ()`
–to eliminate the message queue by freeing the memory.

Create Queue Function

- `MSG_Q_ID msgQCreate (int
maxNumMsg, int maxMsgLength, int
qOptions);`
/ create an ECB pointed by the MSG_Q_ID */*

Insertion and deletion by Queue Send and receive Functions

- msgQSend () Sends into a queue.
(Example 9.24 Step 23)
- msgQReceive() Receives a message into the queue. (Example 9.25 Step 13)

Send and receive functions

- `int msgQReceive (msgQId, &buffer, maxBytes, timeOut)'`

–to let a task wait till sending (posting) of a message by another task.

Wait till either msgQSend function sends the message in a task or till a timeout occurs, whichever happens first.

Options in Queue Functions

- MSG_PRI_NORMAL when the message is sent into the queue for receiving as a FIFO.
- MSG_PRI_URGENT when the message is sent into the queue with this option, the message is received as LIFO

Options in Queue Functions...

- MSG_Q_PRIORITY
 - a task higher in priority than the other waiting ones, takes the message from the queue first.
- MSG_Q_FIFO
 - A task, which first blocked and reached the waiting state, takes the message from the queue first among the waiting ones.

Options in Queue Functions...

- **WAIT_FOREVER**
 - option that can be selected is if wait must be done for sending the message by msgQSend function.
- **NO_WAIT**
 - option that can be selected if no wait to be done. NO_Wait option is simply used for checking the availability of a message

2. POSIX standard Queue functions

POSIX Queue Functions

- mqPxLibInit ()
– to initialize the VxWorks library to permit use of the POSIX Queues.
- mq_open (), mq_close () and mq_unlink ()
– to initialize, close and remove a named queue
- mq_getattr ()
– to retrieves the attribute of a POSIX queue

POSIX Queue ...

- `mq_setattr ()`
 - to set the attribute of a POSIX queue
- `mq_send ()` and `mq_receive ()`
 - to unlock and lock a queue
- function `mq_notify ()`
 - to signal a single waiting task that the message is now available

POSIX Queue ...

- `mq_unlink ()`
 - to unlink but do not destroy the queue immediately but prevents the other tasks from using the queue
- The queue will get destroyed only if the last task closes the queue. Destroy means to de-allocate the memory associated with queue ECB

POSIX Queue Features

- Task notification in case a single waiting task is available.
- There can be 32 message priority levels in place of one priority level URGENT in VxWorks

Summary

We learnt

- queuing of the messages.
- Message can be inserted at back and front
- VxWorks also supports use of POSIX queues.

End of Lesson-12 on VxWorks Queue and Pipe Functions