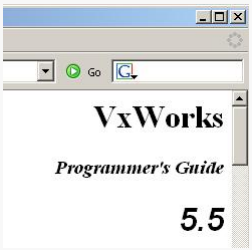
Performance Engineering of Real-Time and Embedded Systems

VxWorks Primitives

VxWorks provides a very rich set of primitives that can appear daunting.

You will probably want to read an overview description of VxWorks.



- 2.2 VxWorks Tasks
- 2.3 Intertask Communications
- 2.4 VxWorks Events
- 2.5 Watchdog Timers
- 3.0 POSIX Standard Interfaces

3

VxWorks v5.5 uses a flat address space in which to execute all tasks.

- VxWorks scheduling centers around tasks
  - ***Tasks execute in the same address space***
  - ***Each task gets its own stack allocation***
  - ***Can use VxWorks or POSIX task primitives.***
- VxWorks provides standard synchronization primitives
  - ***Semaphores, mutual exclusion***
  - ***Message queues***
  - ***Events***

4

There is a variety of activities that you can monitor using WindView.

- Task activities
  - ***Context switching***
  - ***Priority setting***
  - ***Synchronization events***
  - ***…***
- User generated events
  - ***wvEvent***
- Trigger actions to occur when WindView detects an event

5

If you need to change task priorities for the dynamic priority scheduling algorithms…

- This primitive function will be of interest to you.
  - *taskPrioritySet( )* ***Changes the priority of a task.***

If you want to do timing analysis with greater than 60Hz resolution…

- This primitive function will be of interest to you.
  - *sysClkRateSet( )* ***Changes the number of ticks per second.***

6

Watchdog functions execute as interrupt service code.

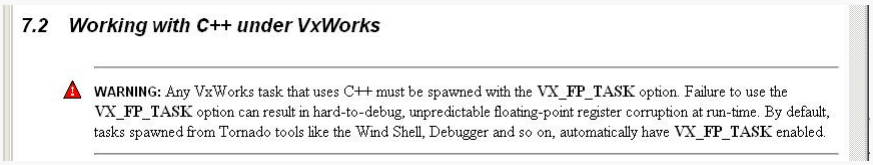
- Watchdog functions run at the interrupt level of the system clock.
  - ***Compare to POSIX timers that send SIGALARM to a task when it is next scheduled.***
- Watchdog functions execute in a context outside of all task contexts.
- Watchdog functions should execute quickly.
  - ***No printfs – use message library***
  - ***No calls which block***

There are limits to what you can do in a watchdog function.

- Watchdog to task communication
  - ***Shared memory***
  - ***semGive except mutual exclusion semaphores***
  - ***Message queues – will discard instead of block***
- Some function calls allowed in watchdogs
  - ***wdStart( ), wdCancel( )***
  - ***logMsg( ), msgQSend( )***
  - ***semGive( )***
  - ***kill( )***
  - ***taskSuspend( ), taskResume( ), taskPrioritySet( ), taskPriorityGet( ), taskIdVerify( ), taskIdDefault( ), taskIsReady( ), taskIsSuspended( )***

You can write VxWorks tasks in C or C++.

- C++ Development in VxWorks
- Heed this warning:



7.2 Working with C++ under VxWorks

WARNING: Any VxWorks task that uses C++ must be spawned with the **VX_FP_TASK** option. Failure to use the **VX_FP_TASK** option can result in hard-to-debug, unpredictable floating-point register corruption at run-time. By default, tasks spawned from Tornado tools like the Wind Shell, Debugger and so on, automatically have **VX_FP_TASK** enabled.

9