

EE 255 Real-Time Embedded Systems

Homework 1

Team05 (Uriel Escobar & Jinwei Zhang)

1. How does a system call execute? Explain the steps from making the call in the userspace process to returning from the call with a result.

Basically the the userspace process invokes a system call function and then the system call jumps into the kernel space. Upon the system call invoked the mode bit is switched to 0. The kernel receives and executes the system call request in the kernel mode. Finally the kernel returns the result of the system call function to the userspace process. After system call been completed the mode bit is set back to 1.

2. When does access to data structures in userspace need to be synchronized?

If there are different threads concurrently executing and accessing to the same critical memory locations, the racing condition could happen. In this case, the synchronization is needed to protect the concurrent threads from undesirably affecting each other's use.

3. What synchronization mechanisms can be used to access shared kernel data structures safely? List them.

mutex,semaphores,spin Locks,no preemption, atomic operations, disabling interrupts

4. Can spinlocks be used on single-processor platforms? Why or why not?

Theoretically they can be used. In practice spinlocks might affect performance since they must wait in a loop until they acquire the lock. A single core processor will see the effects greater as it will halt everything until the locked is acquired. If the wait time is really short then it might not be noticable but if it is not then the user will see a drop in speed.