Embedded Operating Systems

Trainer: Nilesh Ghule

Spinlock

- Hw based sync obj.
- It internally uses bus locking mech to ensure that only one bus op/tx carried out at a time
- This is helpful in multi-processor scenario as well (only one CPU can access bus).
- spinlock var has two states
  - 0 → available (unlocked)
  - 1 → locked.
- pseudo code for spinlock ops
  1. lock var.
  2. init op: lock=0
  3. lock op:
     ```
     while(lock==1);
     lock=1
     ```
  4. unlock op: (by one who locked it)
     lock=0

← if spinlock is already locked the new locking thread keep polling the status. i.e. keep running on CPU. state: busy waiting.

Linux kernel spinlock syntax

1. `#include <linux/spinlock.h>`
2. `spinlock_t lock;`
3. `spinlock_init(lock);`
4. `spin_lock(lock);`
5. `spin_unlock(lock);`

bus locking instruction

In ARM7: spinlock created using SWP instn.
In ARM Cortex: spinlock made using LDREX, STREX, ...

spinlock shold be used only in scenarios where sleep/wait is not allowed e.g. ISR → in device drivers.
spinlocks are available only in kernel space (linux)

Nilesh Ghule

Thank you!

Nilesh Ghule <nilesh@sunbeaminfo.com>