Programming assignment 4

Tao Gong

Lock

- Spin lock
  - While(try_acquire_lock)do_nothing;

- Sleep lock
  - If(cannot_acquire_lock)sleep();
  - Wakeup only when lock is passed to
    - I.e., at release, find 1 thread that is waiting for this lock, and wakeup
  - (Wakeup all waiting threads and let them compete again)

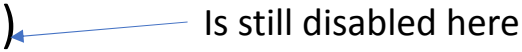
Implementing lock

- Lock data structure <- pthread_mutex_t
  - kern/include/synch.h
- Lock implementation
  - kern/thread/synch.c
  - Lock init
  - Lock acquire
  - Lock release
- Testing
  - Command “sy2”

Atomic access

- spl = splhigh()
  - DisableInterrupts
- (Atomic section)
- splx(spl)
  - RestoreInterrupts

- Can be nested
  - spl = splhigh()
  - spl = splhigh()
  - splx(spl)
  - splx(spl)



Thread sleep and wakeup

- thread_sleep(void*addr)
- thread_wakeup(void*addr)
- (Please read the comments of these functions in thread.c)

- Interrupt must be disabled before calling these functions
- Wakeup or sleep using “addr” as identifier.

- Example:
  - One thread call thread_sleep(1), it will go to sleep state (not scheduled)
  - Another thread call thread_wakeup(1), it will wake the thread called thread_sleep(1)
  - If multiple threads used thread_sleep(1), all of them will wake up.

Implementing sleep lock

- Read and understand wait(P) and signal(V) code
- What's the difference?
  - P <> lock_acquire and V <> lock_release ?

- A lock is similar to a semaphore
- Difference: https://stackoverflow.com/questions/62814/difference-between-binary-semaphore-and-mutex
  - Binary
  - Initial value
  - Ownership

Implementing CV

- CV data structure <- pthread_cond_t
  - kern/include/synch.h
- CV implementation
  - kern/thread/synch.c
  - CV init
  - CV wait
  - CV signal
  - CV broadcast
- Testing
  - Command “sy3”

Implementing CV: basic

- What's the difference?
  - cv_wait <> P and cv_signal <> V ?

- You answered in midterm:
- If no one is waiting,
  - V still counts (1 resource).
  - cv_signal will not.

- cv_broadcast ?

Implementing CV:

- Atomically, do the following: (for cv_wait)
  - Release the lock
  - Put thread to sleep (wake up some threads)
  - Re-acquire the lock
- Similar for cv_signal and cv_broadcast

Demo: spin lock

- Warning: DO NOT implement spin lock in your assignment. You need to implement sleep lock.

Q & A