Assignment 1

Sanzheng Qiao

Department of Computing and Software

February, 2013

Lock

Data structure:

- Name, a string, for debugging
- value, can be BUSY or FREE
- list, for the threads waiting on the lock
- holder, the thread holding the lock

Lock

Lock::Acquire

1. The current thread must not be the lock holder
2. Disable interrupts
3. if the lock is BUSY
   The calling thread sleeps on the waiting list
   else
   Set the lock to BUSY
4. Set holder to the current thread
5. Restore the interrupt level

Lock

Lock::Release

1. The current thread must be the lock holder
2. Disable interrupts
3. Set the lock holder to NULL
4. if the waiting list is not empty
   Remove a thread from the waiting list and make it ready
   else
   Set the lock to FREE
5. Restore the interrupt level

Condition

The data structure

- Name, a string, for debugging
- Queue, for threads waiting on the condition

Condition

Condition::Wait

1. The current thread must be the lock holder
2. Disable interrupts
3. Put the current thread on the wait queue
4. Release the lock
5. The current thread goes to sleep
6. Restore the interrupt level
7. Re-acquire the lock

Condition

Condition::Signal

1. The current thread must be the lock holder
2. Disable interrupts
3. if the wait queue is not empty
   remove a thread from the wait queue and make it ready
4. Restore the interrupt level

Mailbox

Data structure

- A lock for mutual exclusion
- A condition variable for waiting senders to send messages
- A condition variable for waiting receivers for message arrival
- A condition variable for the sender waiting for acknowledgement

Mailbox

Mailbox::Send

1. Acquire the lock
2. While the box is not empty
   wait on the condition waiting senders queue
3. Copy the message to the box
4. Signal a waiting receiver
5. Wait for acknowledgement
6. Signal a waiting sender
7. Release the lock

Mailbox

Mailbox::Receive

1. Acquire the lock
2. While the box is empty
   wait on the condition waiting receivers queue
3. Move the message from the box to the buffer
4. Signal the sender waiting for acknowledgement
5. Release the lock