
Operating Systems: Synchronization

Ali Ataollahi, Ali Abbasi

Overview

Lock



Definition

- Lock or Mutex is a synchronization mechanism for enforcing limits on resources in an environment where there are many threads of execution
 - A lock is designed to enforce a mutual exclusion concurrency control policy.
-

Types of Locks in xv6

The logo for xv6, consisting of the text "xv6" in a black, handwritten-style font, centered within a teal rectangular background.

Spinlock

- Spinlock is a lock which causes a thread trying to acquire it to simply wait in a loop ("spin") while repeatedly checking if the lock is available. Since the thread remains active but is not performing a useful task, the use of such a lock is a kind of busy waiting.

Sleeplock

- Sleeplock does not busy wait, the waiting thread sleeps on the address of the lock, setting the state of the current process to SLEEPING. It won't get scheduled until there is an opportunity to get the lock.
-

XV6 Spinlock

spinlock.h

```
// Mutual exclusion lock.
struct spinlock {
    uint locked;          // Is the lock held?

    // For debugging:
    char *name;           // Name of lock.
    struct cpu *cpu;      // The cpu holding the lock.
    uint pcs[10];         // The call stack (an array of program counters)
                          // that locked the lock.
    int pid;
};
```

XV6 Sleeplock

sleeplock.h

```
// Long-term locks for processes
struct sleeplock {
    uint locked;           // Is the lock held?
    struct spinlock lk;    // spinlock protecting this sleep lock

    // For debugging:
    char *name;            // Name of lock.
    int pid;               // Process holding lock
};
```

Starvation

XV6 locks

They have starvation because the next process which gets the lock is chosen randomly and processes may be starved.

Ticketlock

Another type of lock which implements some level of fairness by keeping a queue of processes in order they have requested to obtain the lock.

Now, Your Project!

Priority Lock

Definition

A lock which can be obtained by priority.

Process want getting locks queueing by priority and at any moment only one process acquiring lock.

Priority Lock

Implementation

Process queueing by their process ID (pid). After a process release lock, process with greatest pid should be dequeued and obtained lock.

Ticket Lock

a synchronization mechanism, or locking algorithm, that is a type of spinlock that uses "tickets" to control which thread of execution is allowed to enter a critical section.

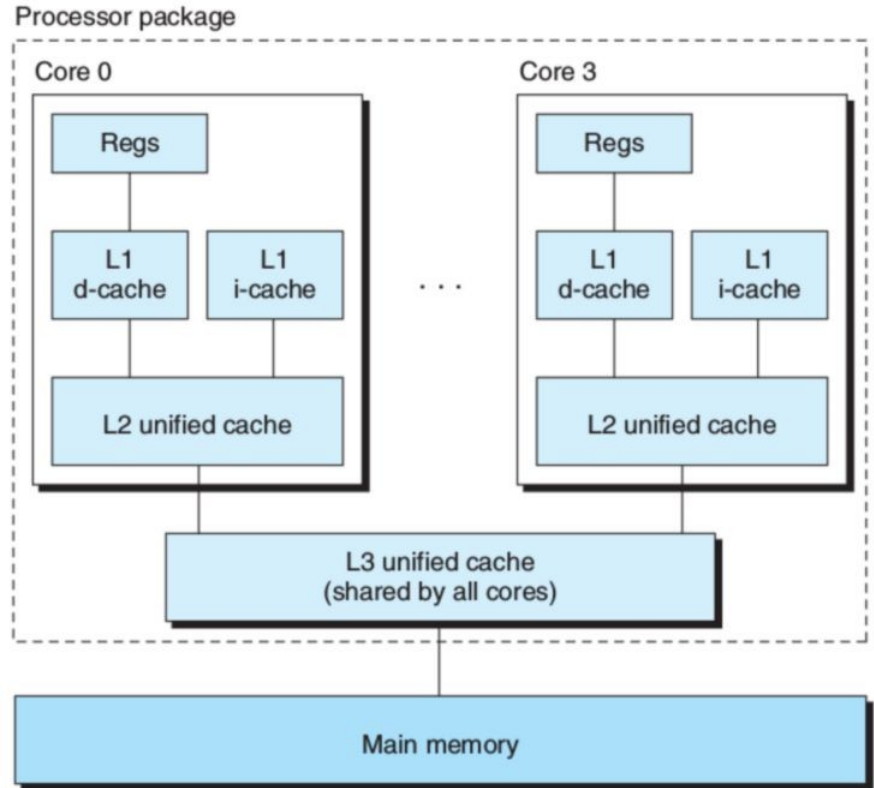
Core i7 Cache Hierarchy

Multiple Levels of Caches

- Local
- Shared

Memory Consistency Models

Cache Coherence Protocols



Coherence Example

An Execution Sequence:

- Assuming X is Shared Between Processes
- Cached Value in Parentheses

Time:
0

Process 1

Core 1
(X = 8)

Print(X)

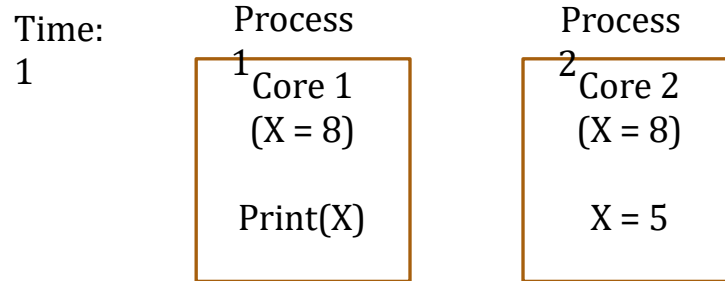
Process 2

Core 2
(X = 8)

X = 5

Coherence Example

An Execution Sequence: (Cont.)



Coherence Example

An Execution Sequence: (Cont.)

Time:
2

Process

¹
Core 1
(~~X = 8~~)

Print(X)

Process

²
Core 2
(X = 5)

X = 5

Coherence Example

An Execution Sequence: (Cont.)

Time:
3

Process

¹Core 1
(X = 5)

Print(X)

Process

²Core 2
(X = 5)

X = 5

Questions?

aliataollahi40@gmail.com

aliabbasi806@gmail.com
