

Задача. D. Ticket Spinlock.

Рассмотрим простой сценарий: в системе исполняются n потоков, каждый из них в бесконечном цикле выполняет короткую критическую секцию. Под короткой критической секцией будем понимать такой фрагмент кода, который можно многократно исполнить в течение одного кванта времени.

Утверждается, что с ростом числа потоков в какой-то момент частота выполнения критических секций (т.е. число критических секций в секунду) в системе значительно упадет.

- 1) При каком числе потоков это случится и почему?
- 2) Почему test-and-set spinlock не подвержен этой проблеме?

Решение. 1) Заметим, что в ticket spinlock если какой-то поток A должен войти(его очередь), но планировщик усыпляет его, то в это время планировщик запускает другие потоки, которые шли за ним, и эти потоки тоже имеют номерки, которые они получили позже потока A , поэтому они тоже будут проверять свое условие вхождения в критическую секцию безуспешно, все будут проверять условия в цикле while и уходить в ожидание.

Теперь, заметим, что в задаче, судя по всему имеется ввиду многоядерная система, а значит у нас есть две несвязанные между собой очереди.

Очередь определенная билетами ticket spinlock и очередь потоков в планировщике, а если планировщик сделает вышеописанную операцию, то при достаточно большом числе потоков(квант времени / (время проверки условия в цикле while + wait)), за квант времени ни один не зайдет в критическую секцию.

Пусть у нас есть N ядер.

Допустим ситуацию, что на каком-то ядре выполняется критическая-секция какого-то потока, и планировщик может сказать этому потоку, что у него закончилось время и сдвинуть его в конец очереди, а в это время поток B стоит в этом ядре на ожидании и планировщик решает отправить поток B отдыхать, так как он уже долго стоит в ожидании, поток B оказывается в конце очереди планировщика, а перед ним находятся потоки с меньшими номерками. Если мы получим такую ситуацию, то будет верно то, что написано выше и мы получим замедление.

Очевидно, что такая ситуация может реализоваться если мы имеем $\geq N + 1$ поток.

И далее она будет повторяться, потому что очереди уже идут в неправильном порядке, и далее по какому-то распределению, количество критических секций, выполняемых в квант времени будет меняться вокруг медленного среднего.

2) В test-and-set spinlock нет билетов, а значит нет двух очередей и он не подвержен этой проблеме, потому что потоки будут исполняться в порядке очереди планировщика, без затрат времени на бесполезные проверки, которые переводят в ожидание.