Locks

Linux-based futex locks

- Mutex lock

mutex < highest bit lock status (31st bit)
remaining bits track no. of waiters of lock

atomic_bit_test_set($\underset{\text{pointer}}{\text{mutex}}$, $\underset{\text{bit}}{31}$) returns previous value

if bit is 0 changes 1, 1 stays 1

if already 1 then lock is with another thread, we increment mutex, since one more waiter now.

Now loop rechecking 31st bit atomic_bit_test set decrement mutex if true and return

Otherwise check if mutex val is non-negative if true that means lock is free, so re loop.

If not futex_wait(mutex, $\underset{\text{expected}}{v}$), mutex should match expected.

- Mutex unlock

  atomic_addzero(mutex, x800...)

  This will result in zero if there are no waiting threads, then return

  Otherwise wakeup one of the them

  futex_wake(mutex)

Flag.s