#3 REWIND_STACK_DO_EXIT

↳ DO_EXIT

↳ EXIT_SIGNALS

↳ MIGHT SLEEP → SEEMS OKAY

PER CPU VARIABLE

#3 and #4 happened because of 7 BUG()

#4 REWIND_STACK_DO_EXIT

↳ DO_EXIT → EXIT_TASK_WORK()

↳ TASK_WORK_RUN

↳ FPUT → AS TASK WORK WHEN EXITING TASK

↳ LOCKS_REMOVE_FILE → SLEEPS

LOCKS_REMOVE_FILE()

↳ LOCKS_REMOVE_FLOCK()

↳ IF NO filp→f_op→FLOCK()

↳ FLOCK_LOCK_INODE()

↳ PERCPU_DOWN_READ_DISABLE()

#1 SYSENTER
- DO FAST SYSCALL
  - EXIT TO USER MODE
    - TASK_WORK_RUN
      - FPUT
        - LOCKS_REMOVE_FILE → SLEEPS

#2 SYSENTER
- DO FAST SYSCALL
  - EXIT TO USER MODE
    - TASK_WORK_RUN
      - FPUT
        - LOCKS_REMOVE_FILE → SLEEPS

ATOMIC_CONTEXT
- SPIN LOCK
- IRQ HANDLER

FILE_BWSEM → PER CPU

FILE_LOCK_LIST → PER CPU

LOCKS_REMOVE_FILE SLEPT IN ATOMIC CONTEXT

↳ TRYING TO ACQUIRE PER CPU SEMAPHORE

IRQ HANDLER ?

- HARD IRQ
  - ENABLED = TRACE_HARDIRQS_ON_THUNK
  - DISABLED = -- SCHEDULE
- SOFT IRQ
  - ENABLED = -- DO_SOFTIRQ
  - DISABLED = CALL_ON_STACK

THEORY:

[redacted]

↳ CALLED INSIDE ATOMIC CONTEXT

FLOCK_LOCK_INODE()

↳ PERCPU_DOWN_READ_PREEMP_DISABLE()

↳ SLEEP

↳ SLEEPS WITH IRQ MASKED

↳ HARD or SOFT

CALLERS

- FLOCK_LOCK_INODE_WAIT()
  - (92) ↳ LOCKS_LOCK_INODE_WAIT()
- LOCKS_REMOVE_FLOCK()
  ↳ LOCKS_REMOVE_FILE
  ↳ __FPUT()

OUR CASE

#1 _NFS4_PROC_SETLK()
#2 LOCKS_LOCK_FILE_WAIT()
#3 NFS4_LOCK_DONE
#4 NFS4_LOCKU_DONE
#5 NFS4_PROC_UNLCK