

## [Lab-12] Interrupt & Kernel synchronization

소프트웨어학과 201921085 곽수정

- **Describe your design and implementation**

먼저 interrupt request handler를 구현하기 위해 my\_driver\_init에 IRQ를 register 하였습니다. IRQ 인터럽트가 발생했을 때 irq\_handler로 인터럽트 서비스 함수를 지정하여 커널에 등록하였습니다. 또한 my\_driver\_exit에는 request\_irq 함수로 등록한 irq\_handler 함수를 제거하기 위해 free\_irq 함수를 사용하였습니다.

multiple processes 에서 struct my\_device\_data 에 있는 shared data 를 synchronization 하기 위해 spinlock 과 mutex 를 생각해볼 수 있었는데, ioctl\_test.c 파일에서 ioctl 함수를 이용하여 디바이스에게 전달한 명령들과 expected answer 를 살펴보았을 때 하나의 process 가 lock 을 잡고 계속 실행해야 원하는 결과를 얻을 수 있을 것이라고 생각되어 mutex 를 사용하게 되었습니다. 부모 process 와 자식 process 둘다 shared data 에 대해 lock 을 잡고 연산하고 print 하기까지의 시간이 길기 때문에 spinlock 보다는 mutex 가 적절하다고 생각하여 사용하게 되었습니다.

따라서 driver.h 에 lock 과 unlock 명령에 대한 새로운 매크로 MY\_IOCTL\_LOCK, MY\_IOCTL\_UNLOCK 을 선언하였습니다. driver.c 에는 먼저 struct mutex mut 을 선언하고 my\_driver\_init 에서 mutex\_init 으로 동적으로 mutex 를 초기화하였습니다. 그리고 MY\_IOCTL\_LOCK 과 MY\_IOCTL\_UNLOCK 커맨드가 들어왔을 때 각각 mutex\_lock 과 mutex\_unlock 을 수행하도록 하였습니다. ioctl\_test.c 에서는 부모 process 에서 ioctl 함수로 MY\_IOCTL\_INC\_0 기능을 수행하기 전에 MY\_IOCTL\_LOCK 으로 뮤텁스를 얻도록 하였고, 마지막 MY\_IOCTL\_RESET 기능을 수행한 이후에 MY\_IOCTL\_LOCK 으로 뮤텁스를 해제하도록 하였습니다. 자식 process 도 마찬가지로 뮤텁스를 얻고 연산을 수행하고 뮤텁스를 해제하도록 하여 multiple process 에서의 shared data synchronization 을 구현하였습니다.

(테스트 결과: 다음장에 첨부)

- Attach the test script result

```
[ 94.603349] IOCTL0, sum: 15
[ 94.603643] IOCTL1, sum: 10
[ 94.611616] IRQ, sum: -15
[ 94.611852] IOCTL2, sum: 25
[ 94.612210] Device File Closed...!!!
[ 94.615027] IRQ, sum: -15
[ 94.615193] IOCTL2, sum: 25
[ 94.617931] IOCTL0, sum: 15
[ 94.618053] IOCTL1, sum: 10
[ 94.618154] Device File Closed...!!!
[ 94.620296] Device File Opened...!!!
[ 94.622394] Device File Opened...!!!
[ 94.623242] Device File Opened...!!!
[ 94.630620] IRQ, sum: -15
[ 94.630955] IOCTL2, sum: 25
[ 94.632067] IOCTL0, sum: 15
[ 94.632248] IOCTL1, sum: 10
[ 94.633679] IRQ, sum: -15
[ 94.634011] IOCTL2, sum: 25
[ 94.634944] IOCTL0, sum: 15
[ 94.635105] IOCTL1, sum: 10
[ 94.635256] Device File Closed...!!!
[ 94.635994] IOCTL0, sum: 15
[ 94.636153] IOCTL1, sum: 10
[ 94.636301] Device File Closed...!!!
[ 94.639138] IRQ, sum: -15
[ 94.639379] IOCTL2, sum: 25
[ 94.648461] Device File Closed...!!!
[ 94.649791] Device File Opened...!!!
[ 94.650263] IOCTL0, sum: 15
[ 94.650360] IOCTL1, sum: 10
[ 94.650933] Device File Opened...!!!
[ 94.651386] IOCTL0, sum: 15
[ 94.651488] IOCTL1, sum: 10
[ 94.653943] IRQ, sum: -15
[ 94.654098] IOCTL2, sum: 25
[ 94.654416] Device File Closed...!!!
[ 94.656982] IRQ, sum: -15
[ 94.657131] IOCTL2, sum: 25
[ 94.657457] Device File Closed...!!!
[ 94.659806] Device File Opened...!!!
[ 94.660435] Device File Opened...!!!
[ 94.663407] IOCTL0, sum: 15
[ 94.663515] IOCTL1, sum: 10
[ 94.666510] Device File Opened...!!!
[ 94.667491] IOCTL0, sum: 15
[ 94.667585] IOCTL1, sum: 10
[ 94.669070] IRQ, sum: -15
[ 94.669224] IOCTL2, sum: 25
[ 94.669529] Device File Closed...!!!
[ 94.675255] IOCTL0, sum: 15
[ 94.675371] IOCTL1, sum: 10
[ 94.676259] IRQ, sum: -15
[ 94.676415] IOCTL2, sum: 25
[ 94.680890] Device File Closed...!!!
[ 94.682230] IRQ, sum: -15
[ 94.682368] IOCTL2, sum: 25
[ 94.682793] Device File Closed...!!!
expected IRQ : -15, [100/100]
expected IOCTL0: 15, [100/100]
expected IOCTL1: 10, [100/100]
expected IOCTL2: 25, [100/100]
[Total Points = [400/400]]
```