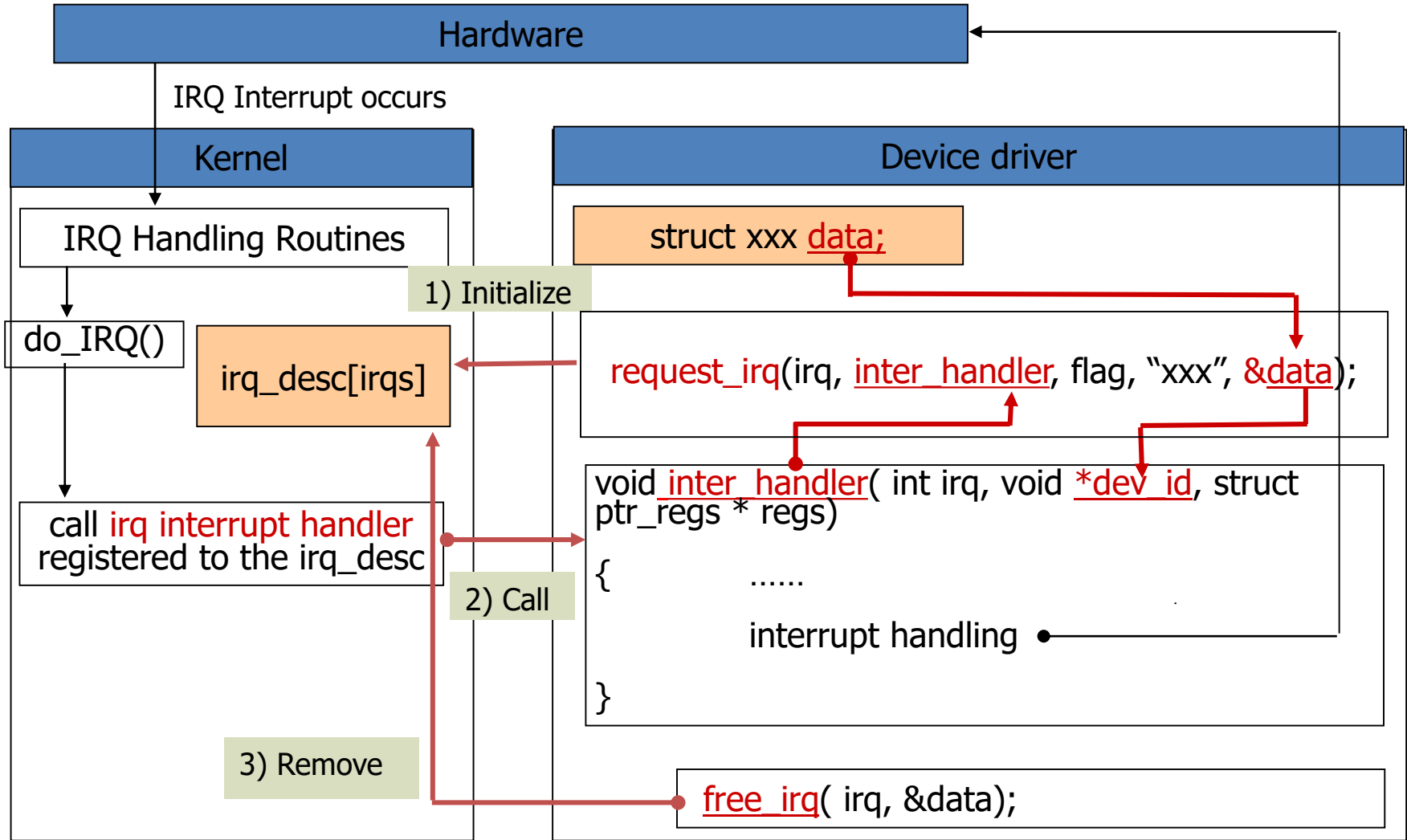


## Interrupt

Dept. of Computer Science and Engineering  
Sogang University, Seoul, KOREA

# Overall Flow of Interrupt Handling



# Kernel 수정

- ➔ **arch/arm/mach-mx6/board-achroimx.c 수정**  
(/work/achroimx\_kernel 내부)

```
188 static struct gpio_keys_button ard_buttons[] = {
189 //  GPIO_BUTTON(SABREAUTO_ANDROID_HOME,    KEY_HOME,        1, "home",        0),
190 //  GPIO_BUTTON(SABREAUTO_ANDROID_BACK,     KEY_BACK,        1, "back",         0),
191 //  GPIO_BUTTON(SABREAUTO_ANDROID_VOLUP,    KEY_VOLUMEUP,    1, "volume-up",    0),
192 //  GPIO_BUTTON(SABREAUTO_ANDROID_VOLDOWN,  KEY_VOLUMEDOWN, 1, "volume-down", 0),
193     GPIO_BUTTON(SABREAUTO_ANDROID_POWER,   KEY_POWER,       1, "power-key",    1),
194 };
```

- ➔ **Kernel 컴파일 및 bootimage 생성**
  - **source /root/.bashrc**
  - **make achroimx\_defconfig**
  - **make -j4**
  - **./make\_bootimage.sh (/work/android에서)**



---

# Kernel 수정

- ➔ **Kernel 기록**
  - **u-boot 모드 진입 (device)**
  - **fastboot (device)**
  - **fastboot erase boot (host)**
  - **fastboot flash boot boot.img (host)**
  - **fastboot reboot (host)**

# Interrupt 실습

## ➔ arch/arm/mach-mx6/board-acroimx.c

```
89 #define SABREAUTO_ANDROID_HOME      IMX_GPIO_NR(1, 11)
90 #define SABREAUTO_ANDROID_BACK      IMX_GPIO_NR(1, 12)

94 #define SABREAUTO_ANDROID_VOLUP      IMX_GPIO_NR(2, 15)

105 #define SABREAUTO_ANDROID_VOLDOWN    IMX_GPIO_NR(5, 14)
```

## ➔ Interrupt Handling

- request\_irq(irq, inter\_handler, **flag**, "xxx", &data)
- 실습에서 쓰는 **Flag**(해당기기에 맞춘것) :  
IRQF\_TRIGGER\_FALLING(누를 때),  
IRQF\_TRIGGER\_RISING(떨 때),  
IRQF\_TRIGGER\_FALLING | IRQF\_TRIGGER\_RISING(누를 때, 떨 대 둘 다)
- void inter\_handler( int irq, void \*dev\_id, struct ptr\_regs \* regs)
- free\_irq( irq, &data);

## ➔ Wait Queue

- wait\_queue\_head\_t
- interruptible\_sleep\_on (wait\_queue\_head\_t \* queue)
- void wake\_up\_interruptible (wait\_queue\_head\_t \* queue)



---

# 기타 tip

- ➡ **Bashrc 다시 적용**
  - `source /root/.bashrc`
- ➡ **Tip : printk 출력 확인(보드-minicom에서)**
  - `echo "7 6 1 7" > /proc/sys/kernel/printk`
- ➡ **C crosscompile**
  - `arm-none-linux-gnueabi-gcc -static -o hello hello.c`
- ➡ **Device file 만들기**
  - `mknod /dev/inter c 246 0`  
(`mknod /dev/[devicename] [type] [major] [minor]`)
- ➡ **전송(host에서)**
  - `adb push 파일이름 /data/local/tmp`