



임베디드 소프트웨어 과제 #2

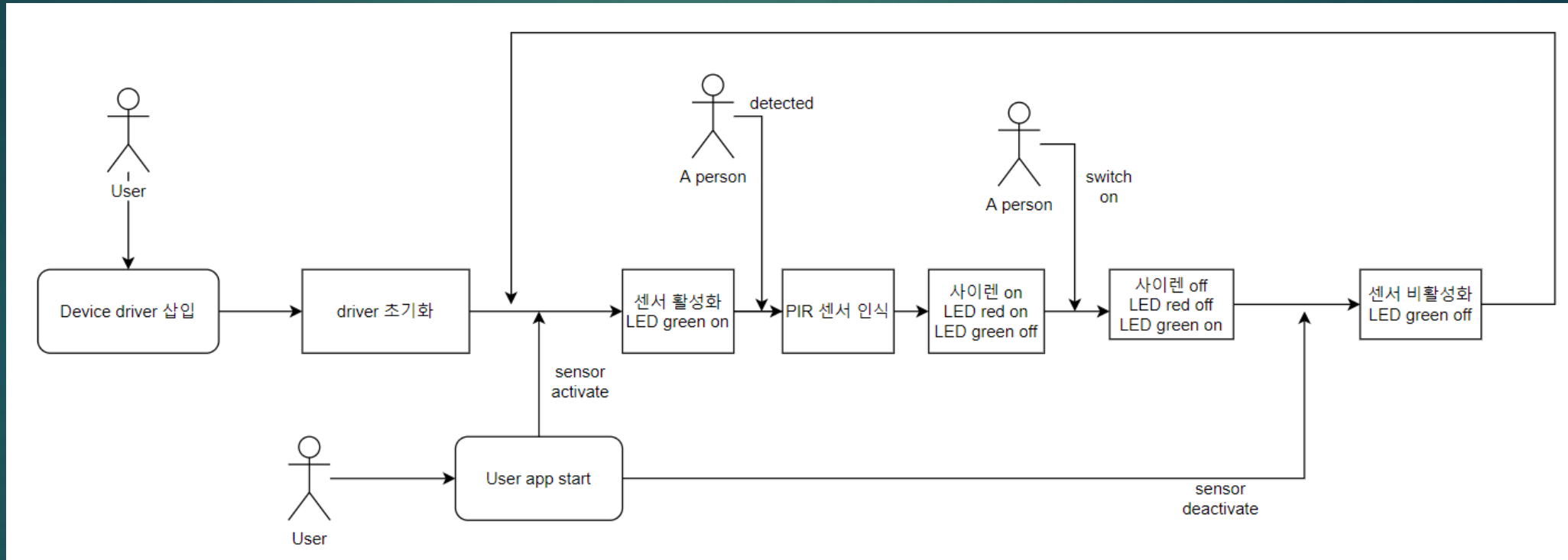
KU_S&A

201511291 컴퓨터공학과 장유준

1. Real world scenario

1. 도둑의 무단 침입을 방지하는 경우
2. 산업 현장에서 위험한 장소에 사람이 있는지 없는지 실시간으로 검사하는 경우

2. Program Flow diagram



3. Used Sensor & Actuator

- 1. PIR sensor – gpio 17
- 2. Switch - gpio 16

**Active
sensors**

- 3. Speaker - gpio 12
- 4. LED red - gpio 5
- 5. LED green - gpio 6

Actuators

4. Struct - device driver

Struct sensor_queue	members	struct list_head list; int arrived_time;
--------------------------------	---------	---

- Sensor가 수신한 시간 값을 저장할 queue 구조체
- sensor가 activated 된 후부터 detect 된 시점을 arrived_time 멤버에 저장

5. Global Variables – device driver

Global Variables	기능
static int irq_num;	PIR sensor의 irq number를 저장할 변수
static int irq_num_switch;	Switch의 irq number를 저장할 변수
static long start_jiffies;	센서 작동 시작 시 초기 시간 값을 저장할 변수
unsigned long time_gap;	PIR 센서의 입력이 발생했을 시 초기 시간 값과의 차이
static struct sensor_queue queue;	Sensor의 수신 시간값을 넣어줄 queue 자료형
static int tasklet_exit;	Tasklet 핸들러가 종료될 시점을 알려줄 변수
static int is_siren_on;	Siren이 시작 됐는지 여부를 판단하는 변수
struct tasklet_struct my_tasklet;	Tasklet 스케줄을 위한 변수
Struct task_struct *my_kthread;	Queue에 있는 값을 전달하기 위한 thread 변수

5. Global Variables – user app

Global Variables	기능
int dev;	Device driver open을 위한 저장 변수
Unsigned long user_buf;	센서 값을 수신하기 유저 버퍼
Pthread_t msgcheck_thread	Thread를 생성하기 위한 thread type 변수

6. Functions – device driver

siren()	Functionality	Play siren
	Parameters	void
	Return Value	void
play()	Functionality	Siren 함수 내부에서 사용 / 해당 노트를 스피커에서 출력
	Parameters	Int note
	Return Value	Void

- Siren() 함수 내부에 play() 함수가 실행되며 스피커에서 사이렌 소리가 나도록 한다

6. Functions – device driver

sensor_isr()	Functionality	PIR 센서 interrupt handler, LED green off, LED red on
	Parameters	int irq void* dev_id
	Return Value	irqreturn_t
tasklet_func()	Functionality	Tasklet handler / sensor 인터럽트 핸들러에서 사용 / queue에 센서값 저장 , 사이렌 소리 출력
	Parameters	unsigned long recv_data
	Return Value	Void

- PIR sensor가 인체를 감지 한 경우 sensor_isr() – 인터럽트 핸들러는 LED green을 off, LED red를 on 한 뒤 tasklet을 스케줄후 종료
- Tasklet handler에서는 queue에 센서값을 저장후 user buffer에 전달, PIR sensor를 disable 한 뒤, cpu core를 점유하며 siren이 1.5초 간격으로 계속 울림 , 사이렌이 종료될 시 sensor의 detect 시점을 유저에게 전달하는 thread를 생성한다

6. Functions – device driver

Send_queue_info ()	Functionality	Queue의 값들을 순회하며 user app으로 전달
	Parameters	Void * args;
	Return Value	void

- Struct task_struct *my_kthread 변수에 연결될 커널 **쓰레드 핸들러**
- Tasklet 핸들러가 종료될 시점에 불리며 그 전까지는 interrupt와 tasklet에 의해 cpu 점유를 하지 못한다 따라서 사이렌이 끝나는 시점에 호출
- Tasklet 핸들러 내부에서 copy_to_user() 함수의 사용은 굉장히 위험하기 때문에 따로 쓰레드를 만들어 전달
- PIR 센서가 detect 될 때의 시간 정보를 user app에 전달
- queue의 값을 dequeue 하며 한 개 씩 user app에 전달
user 역시 마찬가지로 실시간으로 한 개 씩 센서 정보를 전달 받음

6. Functions – device driver

switch_isr()	Functionality	Switch – interrupt handler, LED green on, LED red off, siren off
	Parameters	int irq void* dev_id
	Return Value	irqreturn_t

- PIR sensor가 인체를 감지 한 이후 tasklet이 실행되어 사이렌이 울리고 있는 상태에서 switch를 누르는 경우 인터럽트가 발생
- Switch_isr() – interrupt handle 실행
- Tasklet을 sleep 상태에 들게 하고 tasklet_exit 전역변수를 1로 할당시켜 tasklet이 정상 종료가 될 수 있도록 함 → 사이렌 종료 (사이렌은 switch가 눌리자마자 곧바로 종료되지 않음)

6. Functions – device driver

activate_sensor()	Functionality	PIR sensor irq & switch irq enable, LED green on , LED red off , initiate tasklet , activate sensors
	Parameters	void
	Return Value	void
deactivate_sensor()	Functionality	PIR sensor irq, switch irq disable, LED green off , LED red off , kill tasklet , deactivate sensors
	Parameters	void
	Return Value	void

- IOCTL에 의해 연결되는 함수이며 user app 에서 함수 호출 가능
- PIR센서 인터럽트, switch 인터럽트, LED의 activate를 결정
- Tasklet 초기화

6. Functions – device driver

sa_driver_init()	Functionality	Character device 초기화, 연결, 등록, spin_lock 초기화, gpio 연결, interrupt handler 등록, queue 초기화
	Parameters	void
	Return Value	int
sa_driver_exit()	Functionality	Character device 해제, gpio 연결 해제, queue 제거, interrupt handler 연결 해제
	Parameters	void
	Return Value	int

- 디바이스 드라이버의 초기화, sensor & actuator – gpio 연결
- 디바이스 드라이버의 종료, Queue의 제거, irq_free

7. Functions – user API

activate_sensor()	Functionality	Device driver 내부의 ioctl에서 device driver의 activate_sensor() 함수와 연결, 센서값 수신 thread 생성
	Parameters	unsigned long *user_buff
	Return Value	int
deactivate_sensor()	Functionality	Device driver 내부의 ioctl에서 device driver의 deactivate_sensor() 함수와 연결, thread 제거
	Parameters	void
	Return Value	int

- IOCTL에 의해 연결되는 함수이며 user app 에서 함수 작동
- 센서와 스위치의 activate 여부를 결정, activate 시 LED green 을 on
- 센서의 시간 값을 실시간으로 수신하는 thread를 생성/제거

7. Functions – user API

rcv_msg_thread	Functionality	디바이스 드라이버에서 넘겨주는 센서 값을 확인하며 출력하는 함수
	Parameters	Void *args
	Return Value	Void *

- 디바이스 드라이버의 센서의 수신시간 정보를 실시간으로 전달받으며 전달 받는 값을 출력
- 출력 후에는 다시 user_buf 버퍼 값을 비워줌

8. Demo Video



<https://youtu.be/FpaSdhMYhEE>