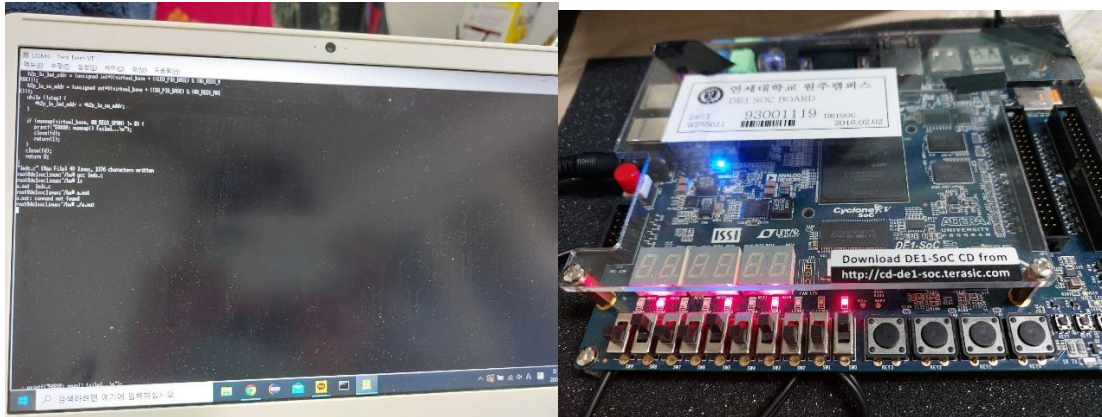


## Hw7

2017253019\_안희영

1. (1) 강의 자료에서 예시된 스위치 상태를 LED에 출력하는 프로그램을 작성하여 수행시키시오. (DE1-SoC 보드의 SD카드에 리눅스가 설치되어 있지 않으면 2번 문제를 먼저 실행하시오.)



스위치가 올라감에 따라 led가 켜지는 것을 확인할 수 있습니다.

- (2) HW\_REGS\_BASE와 HW\_REG\_SPAN의 값이 무엇을 의미하는가? (힌트: DE1-SoC컴퓨터의 메모리 맵을 참고하시오.)

적색 LED용 병렬포트는 base 주소 0xFF200000 에 위치한 데이터 레지스터를 사용합니다. Hw\_reg\_span는 전체 하드웨어 레지스터의 크기입니다..

- (3) mmap 함수의 사용법을 조사하고, 예시 프로그램에서 어떻게 사용하였는지 말하시오

`void *mmap(void *addr, size_t len, int prot, int flags, int fildes, off_t off);`

- addr: 매핑할 메모리 주소
- len: 메모리 공간의 크기
- prot: 보호 모드(or 연산으로 연결하여 사용)
- flags: 매핑된 데이터의 처리 방법을 지정하는 상수
- fildes: 파일 기술자
- off: 파일 오프셋

`mmap( NULL, HW_REGS_SPAN, ( PROT_READ | PROT_WRITE ) , MAP_SHARED, fd, HW_REGS_BASE );`  
맵핑 메모리 주소를 커널이 지정해줍니다. 보호모드는 read 와 write 모드를 사용 합니다.

HW\_REGS\_SPAN만큼의 크기를 맵핑하며 동일파일을 맵핑한 프로세스들이 공유합니다 (map\_shared), 맵핑 시작지점은 HW\_REGS\_BASE입니다.

#### (4) LED\_PIO\_BASE와 SW\_PIO\_BASE는 무엇을 의미하는가?

맵핑된 메모리에 대하여 led 입출력을 담당하는 주소와 스위치 입출력을 담당하는 주소를 오프셋에 대하여 상대적 차이를 상수로 선언하여 접근하기 위한 상대값입니다. 이 경우 오프셋을 led입출력을 담당하는 부분에서 시작했기 때문에 led는 0, 스위치는 0x40의 상대값을 가집니다.

#### (5) led와 sw 장치의 가상 주소를 어떻게 얻었는가?

LED\_PIO\_BASE(led 장치의 주소)와 hw\_reg\_mask의 and 연산값을 통해 맵핑된 메모리 내의 상대값을 계산한 후, virtual\_base(HW\_REGS\_BASE를 시작점으로 맵핑한 가상주소값)에서의 상대적 거리를 더해 led장치의 맵핑 된 가상주소값의 위치를 얻는다. sw장치의 경우 이를 SW\_PIO\_BASE값을 더해 상대적 거리를 얻는다

#### (6) 예시 프로그램을 참고하여 signal 함수의 사용법과 signal handler 등록하는 방법을 설명하십시오. 이 핸들러는 언제 동작하고, 역할은 무엇인가?

```
void (*signal(int signum, void (*handler)(int)))(int)
```

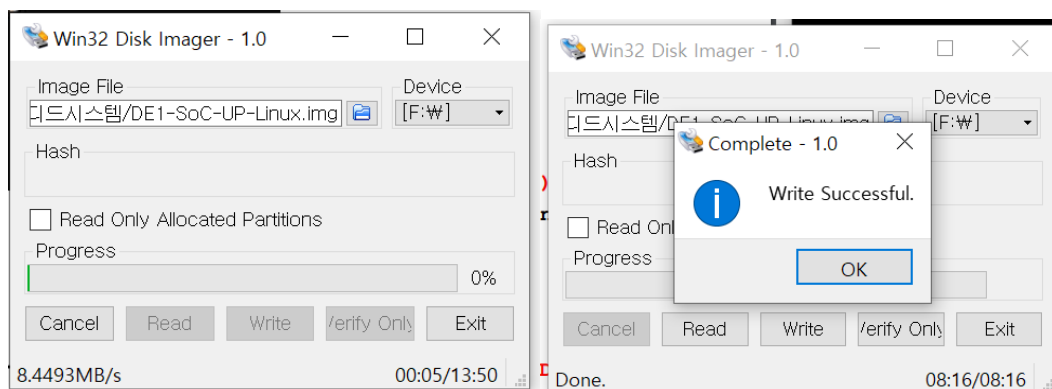
인수: int signum 시그널 번호

oid (\*handler)(int)시그널을 처리할 핸들

SIGINT : ctrl + c 입력발생을 의미하는 시그널 번호

예시 프로그램의 signal 함수는 sigint(종료신호) 인터럽트 발생시 stop을 1로 바꿔주는 catchSIGINT 함수를 인수로 넣어 sigint 발생시 catcSIGINT 함수를 handler로 동작시킨다. signal 함수는 지정된 signal이 발생하면 같이 인수로 들어온 handler 함수를 동작하여 실행한다.

## 2. SD카드에 주어진 리눅스 이 이미지를 설치해보시오



sd카드에 리눅스 이미지를 설치, 1번을 진행했습니다.