**Embedded System Software 과제 2**

**(과제 수행 결과 보고서)**

**과목명: [CSE4116] 임베디드시스템소프트웨어**

**담당교수: 서강대학교 컴퓨터공학과 박 성 용**

**학번 및 이름: 20171670, 이재훈**

**개발기간: 2022. 05. 04. - 2022. 05. 17.**

**최 종 보 고 서**

**I. 개발 목표**

- 각 과제마다 주어지는 주제를 바탕으로 본 과제에서 추구하는 개발 목표를 설정하고 그 내용을 기술할 것.

* 두 개의 IOCTL 명령어를 사용하여 새로운 타이머 디바이스 드라이버 및 관련 응용 프로그램을 개발한다. 응용 프로그램에서 타이머 디바이스 구동 옵션을 받아 ioctl을 통해 디바이스 드라이버에 전달한다.
* Ioctl을 통하여 전달 받은 옵션들을 각 디바이스의 요구사항에 따라 동시에 출력한다.

**II. 개발 범위 및 내용**

- 자신들이 설계한 개발 목표를 달성하기 위하여 어떠한 내용의 개발을 수행할 지 그 범위와 개발 내용을 기술할 것.

**가. 개발 범위**

**1. 응용 프로그램**

**-** ./app TIMER\_INTERVAL[1-100] TIMER\_CNT[1-100] TIMER\_INIT[0001-8000] 와 같은 양식으로 입력을 받는다.입력 받은 옵션들을 타이머 디바이스 드라이버가 구동할 수 있도록 전달한다. Timer\_interval은 HZ값을 의미하고 Timer\_cnt의 경우 디바이스에서 출력 변경 횟수를 의미하며 Timer\_init은 FND에 출력되는 초기 문양과 위치를 말한다. Timer\_init은 4자리 중에서 단 하나만 1부터 8사이 숫자 값을 입력받을 수 있도록 한다.

**2. 타이머 디바이스 드라이버**

**-** FND에서는 TIMER\_CNT횟수만큼 TIMER\_INTERVAL마다 다음 숫자를 지정된 위치에 출력을 한다. 문양의 경우 처음 TIMER\_INIT에서 0이 아닌 숫자이며 다음 문양의 경우 현재 수에서 1증가된 수를 의미한다.

- LED는 현재 FND에서 출력 되고 있는 문양의 번호를 나타낸다.

- TEXT\_LCD에서 첫번째 줄에는 저의 학번을, 두 번째 줄에는 저의 이름을 영문으로 출력한다. TIMER\_INTERVAL마다 각각 오른쪽으로 한칸씩, 만약 끝에 다다를 경우 반대 방향으로 이동합니다.

- DOT에서는 현재 FND에서 출력되고 있는 문양을 출력합니다

- TIMER\_CNT가 0으로 될 경우 모든 것을 0으로 초기화하고 LCD는 빈 칸으로 초기화합니다.

**나. 개발 내용**

1. 응용 프로그램

- 텍스트이(가) 표시된 사진

자동 생성된 설명

위와 같은 구조체를 선언 후 사용자로부터 받은 데이터를 목적에 맞게 구조체에 대입한다. 사용자로부터 받은 데이터가 양식에 맞추어 들어온건지 확인을 진행 한 후 옵션에 맞게 들어온다면 ioctl을 사용하여 타이머 디바이스 드라이버로 데이터를 넘겨준다. 넘겨준 후 다른 ioctl을 사용하여 timer을 실행시켜준다.

2. 타이머 디바이스 드라이버

- 데이터를 넘겨주는 ioctl이 올 경우 타이머 디바이스 드라이버 내 변수들을 초기화 및 데이터를 삽입한다. 그 후 timer를 실행시키는 ioctl이 올 경우 add\_timer함수를 통하여 타이머 디바이스를 실행시킨다. 그 후 주어진 문자와 조건에 맡게 데이터를 변경하고 출력한다.

**III. 추진 일정 및 개발 방법**

- 자신들이 설정한 개발 목표를 달성하기 위한 개발 일정을 설정하고, 각 요소 문제를 해결하기 위해서 어떤 방법을 사용할 지 기술할 것.

**가. 추진 일정**

**05.04~05.06 설계 및 ioctl 숙지**

**05.07~05.13 디바이스 드라이버 구현**

**05.13~05.14 응용 프로그램 구현**

**05.15~05.17 보고서 작성**

**나. 개발 방법**

**1. 응용 프로그램**

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

받은 데이터의 양식이 다를 경우 ioctl로 넘겨주지 않고 에러를 출력하며 return한다.

IOCTL의 경우 데이터를 넘겨줄 수 있는 \_IOW와 타이머 시작을 위한 \_IO를 설정하였다.

- 텍스트, 스크린샷, 화면이(가) 표시된 사진

자동 생성된 설명

만약 데이터를 넘겨주는 ioctl 명령이 들어올 경우 copy\_from\_user를 통하여 데이터를 넣어준다. Name\_i와 name\_dir의 경우 lcd에서 이름의 위치와 방향을 의미하고 num\_i와 num\_dir의 경우 학번의 위치와 방향을 의미한다. Value의 경우 현재 출력되는 fnd데이터를 의미한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

넘겨 받은 FND 데이터 중 1이 아닌 수를 가져온다. 만약 그 수가 8을 넘을 경우 에러인 것을 출력 후 return한다. 올바르게 들어왔을 경우 lcd를 학번과 이름으로 초기화를 해주고 display를 해줍니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

디스플레이 함수에서는 각 조건에 맞춰서 dot, led, fnd, lcd에 출력을 한다. Dot과 led의 경우 현재 fnd에서 출력되고 있는 숫자를 출력하고 lcd의 경우 학번과 이름을 time\_interval에 따라 출력한다. 출력은 outw함수를 사용하여 출력한다.

먼저 fnd에 관해 설명을 하겠습니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Fnd는 time\_interval마다 수가 증가합니다. 만약 8을 넘을 경우 1로 다시 돌아갑니다. 만약 한 사이클을 다 돌 경우 오른쪽으로 이동해서 출력을 합니다. 만약 맨 오른쪽에서 출력하고 있고 이동해야하는 경우 다시 맨 왼쪽으로 이동하여 출력을 진행합니다. 이 때 value는 현재 fnd에서 출력되고 있는 수를 담고 있고 init\_fnd는 초기에 0이 아닌 숫자가 무엇인지 저장하고 있습니다. Pos는 현재 init\_fnd가 fnd에서 출력되고 있는 위치입니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

먼저 value값을 1 올려줍니다. 만약 올린 value값이 초기의 0이 아닌 fnd값과 같을 경우 위치를 옮긴후 출력을 해줍니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

타이머의 경우 cnt와 time\_interval이 있습니다. Cnt는 timer\_cnt의 -1값으로 초기화를 해줍니다. 처음부터 time\_interval만큼 출력을 해줘야하기 때문입니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Lcd에 관한 변수입니다.

텍스트, 오렌지, 어두운이(가) 표시된 사진

자동 생성된 설명

길이측정을 위한 학번과 이름의 길이입니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

학번을 움직이는 코드입니다. Timer\_interval에 따라서 움직입니다. 만약 오른쪽 끝에 도달할 경우 왼쪽으로 이동하고 다시 왼쪽 끝에 도달할 경우 오른쪽으로 이동하게 됩니다. 방향은 left는 0으로 right는 1로 간주하여 이동합니다. Num\_i는 현재 학번 첫문자의 위치고 num\_dir은 진행해야하는 방향입니다.

텍스트, 모니터, 화면, 스크린샷이(가) 표시된 사진

자동 생성된 설명

밑에서 출력하는 이름 또한 학번과 같은 조건으로 설정한다. 다른 점은 text\_lcd의 16번째부터가 밑에서 출력 되는 것이기에 왼쪽끝을 16 오른쪽 끝을 32로 보고 설정합니다.

텍스트, 화면, 스크린샷이(가) 표시된 사진

자동 생성된 설명

단위가 10분의 1초기 때문에 HZ에 10을 나눈 후 time\_interval을 곱해줍니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

만약 cnt가 0이 될 경우 초기화를 하고 display를 해줍니다.

텍스트이(가) 표시된 사진

자동 생성된 설명

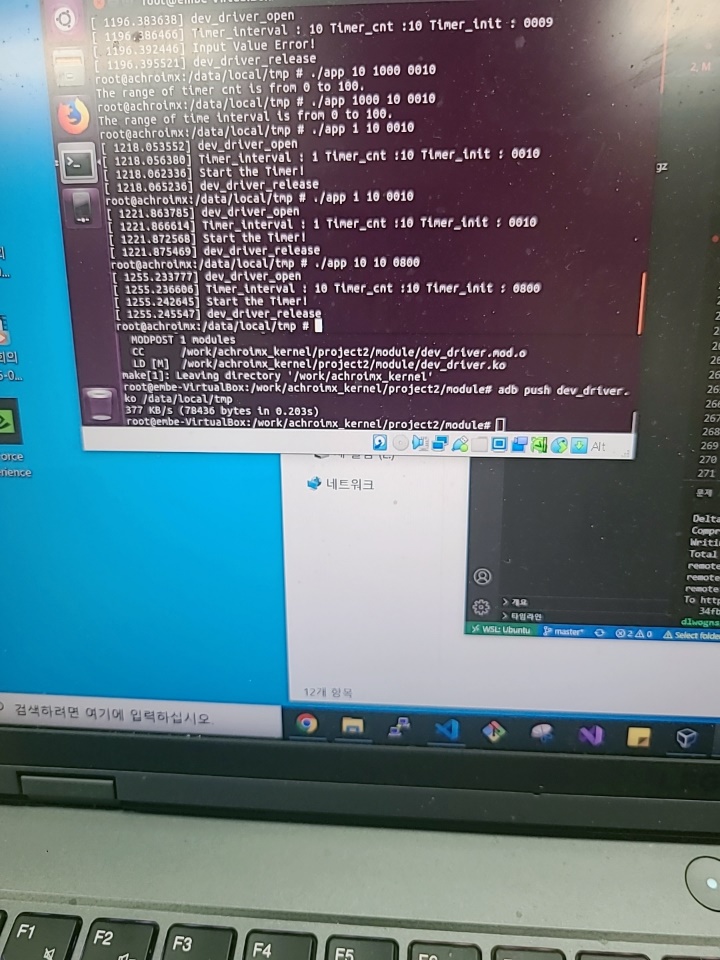
Init과 ioremap을 사용하여 각 디바이스에 맞게 할당을 해줍니다. exit함수에서는 할당한 디바이스 드라이버를 iounmap을 통하여 해제합니다.

**IV. 연구 결과**

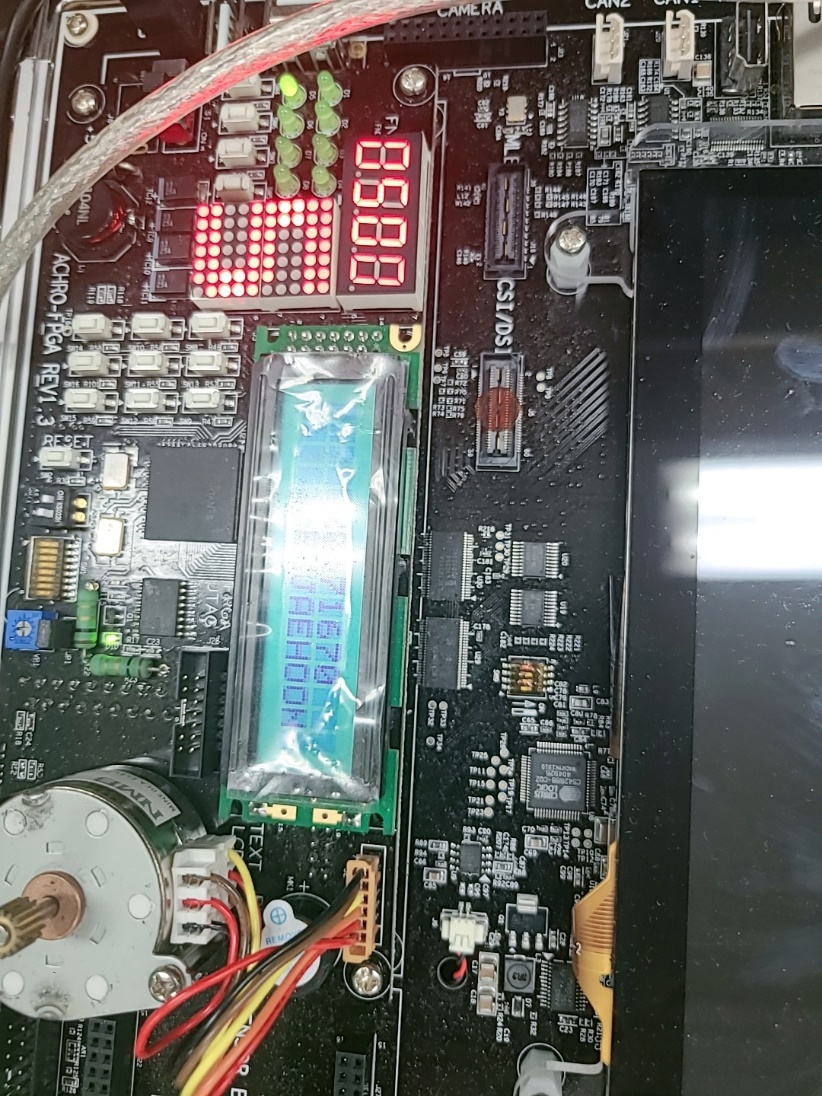
**실행방법**

* **각 폴더 내에서 make을 합니다.**
* **Module 에서는 dev\_driver.ko를 App에서는 app을 /data/local/tmp로 보내줍니다.**
* **Minicom 내에서 insmod dev\_driver.ko를 합니다.**
* **Mknod /dev/dev\_driver c 242 0 을 하여 디렉토리를 만들어 줍니다.**
* **./app 에서 옵션에 맞게 input을 넣어줍니다.**

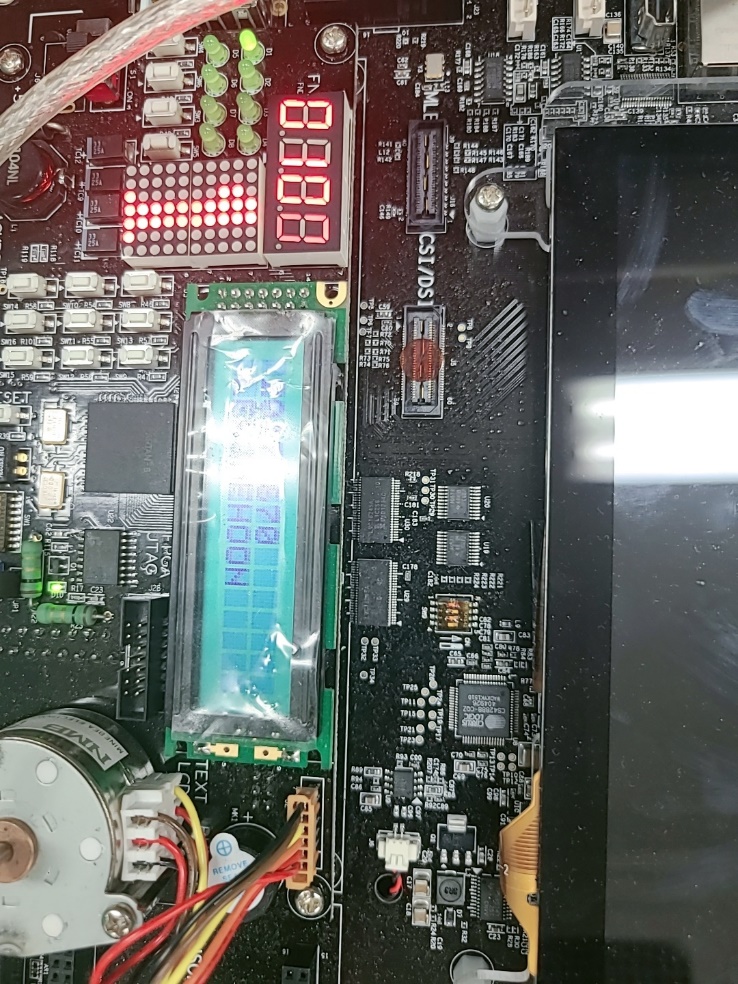
- 최종 연구 개발 결과를 자유롭게 기술할 것.



실행시켰을 때 콘솔 창입니다.



실행 시키고 중간 과정입니다.



중간과정 두번째 입니다.



끝나고 난 후의 사진입니다.

**V. 기타**

- 본 설계 프로젝트를 수행하면서 느낀 점을 요약하여 기술하라. 내용은 어떤 것이든 상관이 없으며, 본 프로젝트에 대한 문제점 제시 및 제안을 포함하여 자유롭게 기술할 것.

지금까지 실습에서도 ioctl에 관한 경험이 없었는데 이번 기회에 찾아보고 쓰면서 어떤 방식으로 사용하는지 어떠한 것들이 있는지 알 수 있었습니다. 또한 제가 직접 몇 개의 device\_driver를 종합하여 사용하고 출력해봄으로써 어떻게 각 device로 연결이 되는지, 어떤 식으로 출력하고 입력을 받고 소통을 하는지 알 수 있었습니다.

실습만으로 module과 device\_driver에 대한 개념을 확실히 잡기 힘들었는데 이번 프로젝트를 통해 확실히 잡을 수 있었고 이 것을 기반으로 다른 개념들 또한 이해할 수 있었습니다.