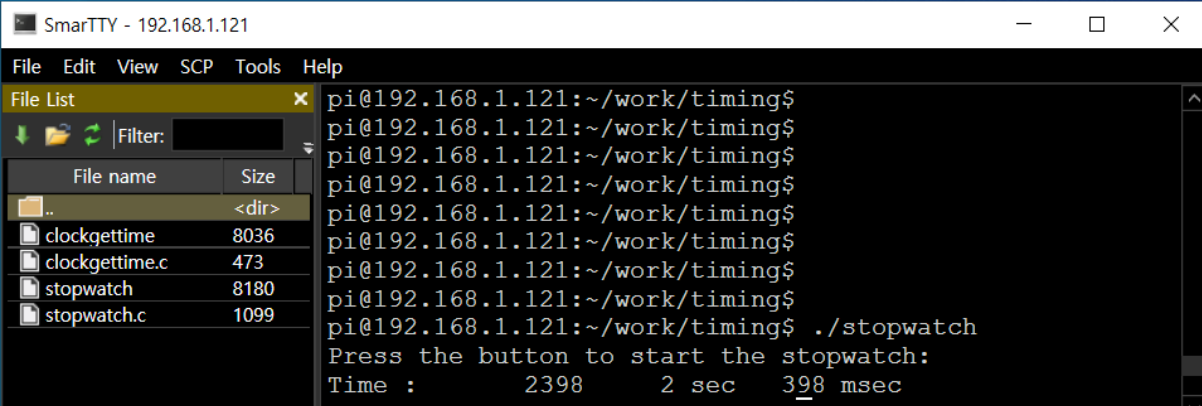


Lab4

이번 실습 과제는 응용 프로그램과 함께 드라이버 소스도 수정해서 구현하는 과제입니다. 라즈베리파이의 /home/pi/work/timing 디렉토리의 stopwatch 프로그램을 실행해 봅니다. 이 프로그램은 버튼을 사용하므로 gpio_button 디렉토리에서 load.sh을 실행해서 버튼 드라이버를 설치한 후 프로그램을 실행해야 합니다.



```
SmarTTY - 192.168.1.121
File Edit View SCP Tools Help
File List
Filter:
File name Size
.. <dir>
clock_gettime 8036
clock_gettime.c 473
stopwatch 8180
stopwatch.c 1099
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$
pi@192.168.1.121:~/work/timing$ ./stopwatch
Press the button to start the stopwatch:
Time : 2398 2 sec 398 msec
```

stopwatch.c 소스를 활용하여 아래의 기능을 구현합니다.

(1) 시간을 LCD 디스플레이의 첫 줄에 표시하십시오. 표시 형식은 분, 초, 1/100 초를 6자리의 고정된 숫자 위치로 나타내고 숫자 사이는 콜론(:)으로 구분하십시오. 예를 들어서 1분 23초 45/100초라면 01:23:45 와 같이 나타냅니다.

(2) 주어진 스톱워치 소스는 왼쪽 버튼 만을 사용하지만, 이 과제에서는 추가로 오른쪽 버튼을 사용합니다. 오른쪽 버튼의 용도는 시간 clear 입니다. 즉, 스톱워치의 현재 상태와 상관 없이 오른쪽 버튼을 누르면 시간은 00:00:00 으로 clear 됩니다. 스톱워치가 정지된 상태에서 clear 버튼을 누르면 00:00:00 을 표시하고 정지된 상태를 유지합니다. 스톱워치의 시간이 진행 중에 clear 버튼을 누르면 00:00:00에서부터 시간을 계속 진행합니다. 버튼 드라이버는 한 개만 사용합니다. 즉, 주어진 버튼 드라이버의 소스를 수정하여 2개 버튼을 다 읽을 수 있도록 수정합니다. 오른쪽 버튼 입력은 GPIO 24 입니다. (버튼 드라이버를 수정하기 전에 주어진 드라이버 소스에서 GPIO를 24로 바꾸어서 오른쪽 버튼을 테스트를 해 보십시오.) 변경된 드라이버를 메인 프로그램에서 사용하기 위해서 전달해야 하는 변수 등의 사항은 본인이 결정합니다. 보고서에는 본인이 어떤 방법으로 어떤 데이터를 전달하도록 드라이버 소스를 수정했는지 명기합니다.

(3) LCD 디스플레이의 두번째 줄을 이용하여 현재 스톱워치의 상태를 나타냅니다. 정지된 상태에서는 Stopped, 스톱워치가 진행 중이라면 Running, clear 버튼을 눌렀을 때는 Cleared 의 메시지가 나타납니다. Cleared의 메시지는 다른 버튼을 눌러서 상태가 변경될 때까지 그대로 유지합니다. 즉, 스톱워치가 진행 중에는 Running의 메시지가 보이지만, clear 버튼을 누르면 Cleared로 바뀌고 왼쪽 버튼을 눌러서 Stopped 상태가 될 때 까지는 Cleared의 메시지가 그대로 남아 있습니다. 즉, 메시지는 버튼을 누를 때만 바뀝니다.

참고 사항: 주어진 LCD 드라이버는 첫 줄 만을 사용합니다. 두 줄을 모두 사용하기 위해서는 응용 프로그램에서 표시 줄(첫번째 줄, 두번째 줄)을 변경하기 위한 방법이 필요합니다. 이 방법은 여러 가지가 가능하지만 본인이 고안해서 구현합니다. 한가지 방법은 드라이버의 ioctl 함수를 추가하여 표시 라인을 변경하는 방법입니다. 구현하기 위해서 필요한 정보는 본인이 검색을 통해서 파악해야 합니다. 아래의 사이트가 비교적 정리가 잘 된 사이트입니다.

<https://www.lazenca.net/pages/viewpage.action?pageId=23789739>

보고서 제출 시 드라이버와 테스트 프로그램의 소스 파일을 첨부하십시오.

위의 3가지 과제 중 3가지를 모두 구현하지 못했을 경우에는 몇 번 과제까지 구현해서 제출하는지 보고서에 명시해 주세요.