

2023.06.24.

AIoTSoftware

Term Project report

2020105580 고예찬

2018102161 김건

요약

cgi와 apache2서버를 이용해서 html서버를 구현한다.
wiringPi로 FND에서 시간을 표현하는 0.01초 스톱워치를 구현한다.
pthread를 이용해서 멀티스레드를 구현하고 각 함수에서
클라이언트에서 전달받은 정보에 따라서 스톱워치의 상태를 변형한다.

목차

- I . 코드
- II . 시연영상

I . 코드

본 프로그램의 코드는 3개의 c코드와 1개의 html코드로 이루어져 있습니다. 먼저 stopwatch.html은 3개의 버튼, start버튼, stop버튼, clear버튼으로 이루어져 있습니다. 각각의 버튼은 서버에게 “Start”, “Stop”, “Clear”라는 정보를 전달하게 됩니다.

다음으로 stopwatch.c와 stop.c, clear.c입니다. stopwatch.c는 wiringPi와 pthread를 이용해서 클라이언트의 조작으로 실행되는 stopwatch와 멀티스레딩을 통해 클라이언트로부터 얻은 정보를 처리하는 것을 포함하는 함수입니다. stop.c와 clear.c는 클라이언트의 조작을 처리하는 함수입니다. 각각의 코드는 stopwatch.html의 3개의 버튼, start(stopwatch)버튼, stop버튼, clear버튼에 대응되는 start.cgi, stop.cgi, clear.cgi로 컴파일되고 stopwatch.html과 함께 서버를 구성하게 됩니다.

1. stopwatch.c

stopwatch.c는 main(), FndSelect(), FndDisplay(), fndThread(), fifoThread(), Init()으로 이루어져 있습니다. 여기서 FndSelect()와 FndDisplay()는 강의시간의 것과 유사하게 작동하면서 FND에 시간을 표시하는 함수입니다. fndThread()와 fifoThread()는 각각 pthread에 의해서 멀티스레드로 작동하는 함수입니다. Init()의 경우 wiringPi의 gpio를 설정해주는 함수입니다.

main()은 2가지의 역할을 수행합니다. 하나는 처음 실행됐을 때, 스톱워치와 멀티스레딩을 준비하는 역할, 둘은 처음 실행이 종료되고 클라이언트로부터 “Start”정보를 받아오는 역할입니다. 스톱워치와 멀티스레딩을 준비하는 단계에서는 먼저 FIFO파일을 생성합니다. 해당 파일에 클라이언트로 전달받은 정보인 “Start”, “Stop”, “Clear”를 저장하게 됩니다. 그리고 fork()를 통해서 자식프로세스를 생성합니다. 자식프로세스에서는 pthread를 이용해

fndThread()와 fifoThread(), 두 함수로 스레드를 생성하고, FIFO파일에 Start를 출력하고 종료됩니다. 이후에 stopwatch.c가 실행될 때는 클라이언트에서 start버튼이 눌러졌을 때로 FIFO파일에 “Start”를 출력하는 역할만 수행하게 됩니다.

fifoThread()는 멀티스레딩으로 작동하는 함수입니다. 프로그램이 실행되는 동안 while문을 계속해서 돌면서 FIFO파일을 열고 전역변수인 status를 start상태일 때 1, stop상태일 때 0, reset(clear)상태일 때 -1을 변경합니다.

fndThread()도 멀티스레딩으로 작동하는 함수입니다. 프로그램이 실행되는 동안 while문을 계속해서 돌면서 전역변수인 status에 상태에 따라서 0.01초 스톱워치를 조작합니다. status=1이면 스톱워치가 작동하고, status=0이면 멈추고 status=-1이면 break;으로 while문을 나가서 FIFO파일을 삭제합니다. 이후 다시 클라이언트의 start버튼이 눌러지면 main()이 처음부터 작동하게 됩니다.

FndSelect(), FndDisplay()는 앞서 설명했던 강의자료의 것과 거의 동일합니다. 단 분, 초, 소수점 이하 초를 구분하기 위해서 2, 4번 위치에서 0x80를 or연산을 통해서 점을 출력하도록 구현했습니다.

2. stop.c, clear.c

stop.c와 clear.c는 클라이언트에서 버튼이 눌러졌을 때, 실행되고 각 버튼에 맞는 Value를 FIFO파일에 출력하는 구조입니다. 만약 FIFO파일이 존재하지 않거나 FIFO파일로의 출력이 성공적이지 못할 때를 확인할 수 있도록 구현되어 있습니다.

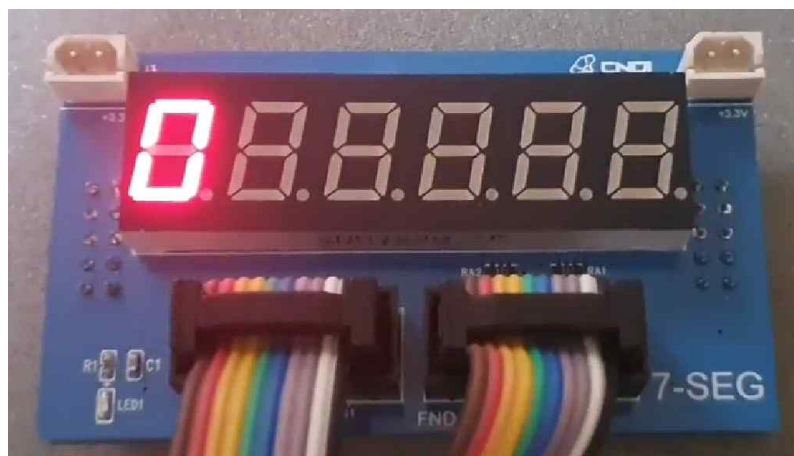
II. 시연영상



start



stop



clear