**OS Project Report 1: Alarm Clock**

20205031 김소정, 20205135 이수린, Team F(team 5)

1. **Problem definition**

현재 Pintos 운영체제의 timer\_sleep() 함수는 busy-waiting 방식을 사용하여 구현되어 있다.이 timer\_sleep()은 잠들어야 하는 ticks까지 도달했는지 while()문으로 계속해서 확인하고, 만약 도달하지 못했으면 thread\_yield()를 호출하여 현재 스레드를 다시 ready\_list로 넣고 다른 스레드를 scheduling해서 불러온다. 그러나 이런 방식은 잠든 스레드도 다시 스케줄링될 수 있도록 만드므로 새로 불린 스레드도 잠들어 있을 수 있다. 만약 이 양보받은 스레드도 아직 깨어날 시간이 아니면 다시 다른 스레드에게 양보를 한다. 이런 방식은 불필요한 context switching이 많이 일어나므로 CPU 자원을 낭비하며 비효율적이다.

따라서 busy-waiting을 피하기 위해 잠든 스레드는 스케줄링되지 않도록 하고, 특정 시간이 지나면 깨우는 방식을 채택하고자 한다.

1. **Policy and algorithm design**

위의 문제를 해결하기 위해 잠든 스레드는 BLOCKED 상태로 전환하여 ready list에서 scheduling되지 않도록 변경할 것이다. 이를 위해 스레드를 재우는 thread\_sleep()과 일정 시간이 경과하면 스레드를 깨우는 thread\_wake()를 사용하여 timer\_sleep()을 수정할 것이다.

* Policy
  + timer\_sleep() 함수에서는 더 이상 매번 스레드의 차례를 미루지 않고, 대신 잠든 스레드를 block하여 schedule되지 않도록 함
  + block될 때 일어나야 하는 시간을 wakeup\_time이라는 thread structure의 새로운 변수에 저장하고 ready\_list가 아닌 sleep\_list에 추가
  + 타이머 인터럽트가 발생할 때마다, 현재 시간과 일어나야 할 시간을 비교하여 스레드를 깨우고 sleep\_list에서 ready\_list로 옮겨 schedule될 수 있도록 함
* Algorithm design
  + Step (1): timer\_sleep()에서 while 루프를 삭제하고 thread\_sleep()을 실행한다. 이 함수는 현재 스레드를 BLOCKED 상태로 전환하며 스레드가 깨어나야 하는 시간을 함께 저장한다.
  + Step (2): 주기적으로 발생하는 타이머 인터럽트가 발생하면, thread\_wake()가 실행한다. sleep\_list의 BLOCKED 상태에 있는 스레드 중에서 현재 시간이 wakeup\_time 이상인 스레드를 ready\_list로 옮긴다.

1. **Mechanism**

* ADD: thread\_sleep 함수, thread\_wake 함수
* Modified file names
  + /home/name/pintos/src/threads/thread.c
  + /home/name/pintos/src/devices/timer.c
  + /home/name/pintos/src/threads/thread.h
* CODES
  + timer\_sleep()

A screen shot of a computer program

Description automatically generated

While()로 불필요하게 context switching을 하는 대신 thread\_sleep()을 부른다.

* + 추가된 thread 구조체 변수 wakeup\_time

A screenshot of a computer program

Description automatically generated

스레드가 일어나야 할 시간을 저장하는 wakeup\_time을 만들어 준다.

* + thread\_sleep()

A computer screen shot of a program

Description automatically generated

스레드를 잠들게 하는 함수로, 실행하는 동안에는 인터럽트를 off해야 하며 실행이 끝나면 다시 인터럽트를 허용한다. 이 때 idle 스레드는 잠들면 안 되므로 assert 처리해준다. 현재 스레드의 wakeup\_time에 일어나야 할 시간 (현재 시간+잠들 시간)을 넣고 block하여 scheduling될 수 없도록 한다.

* + thread\_wake()

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

위의 코드는 sleep\_list에서 일어나야 할 시간이 된 스레드를 깨우는 함수이다. 이 함수는 sleep\_list를 순회하면서 각 스레드의 wakeup\_time과 현재 시간을 비교하여 일정 시간이 경과한 스레드를 깨우고, 해당 스레드를 sleep\_list에서 제거하고 unblock하여 ready\_list로 옮긴다. 그렇지 않은 경우 다음 노드로 이동한다.

* + timer\_interrup()

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

이 함수는 타이머 인터럽트가 발생할 때마다 호출되며, 현재 시간을 기록하고, 스레드 스케줄링을 수행한 뒤 BLOCKED 스레드를 깨우는 역할을 한다.