



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0053157
(43) 공개일자 2021년05월11일

(51) 국제특허분류(Int. Cl.)
H04L 12/863 (2013.01) H04L 12/801 (2013.01)
H04L 29/08 (2006.01)
(52) CPC특허분류
H04L 47/622 (2013.01)
H04L 47/193 (2013.01)
(21) 출원번호 10-2020-0074198
(22) 출원일자 2020년06월18일
심사청구일자 2020년06월18일
(30) 우선권주장
1020190138702 2019년11월01일 대한민국(KR)

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
이원준
서울특별시 강남구 선릉로 221, 104동 2005호 (도곡동, 도곡테크아파트)
김순재
서울특별시 성북구 고려대로7가길 73, 301호 (안암동2가)
(74) 대리인
특허법인엠에이피에스

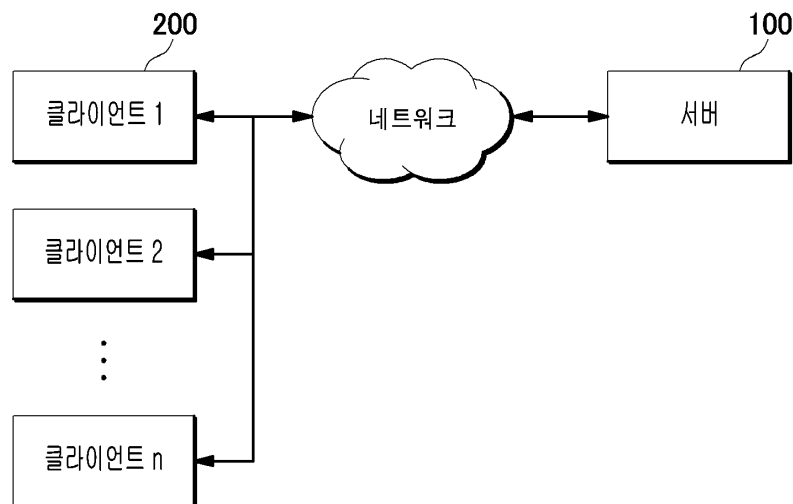
전체 청구항 수 : 총 13 항

(54) 발명의 명칭 스트림 스케줄링 방법 및 서버

(57) 요약

본원의 일 측면에 따른 서버에 의해 수행되는 스트림 스케줄링 방법은, 클라이언트로부터 데이터 전송 요청이 수신되면, 기설정된 조건에 따라 데이터를 복수의 스트림에 할당하는 단계; 스트림에 할당된 데이터의 크기와 요청 시간을 측정하는 단계; 및 스트림 대기열에 스트림을 배치하고, 측정된 데이터 크기 및 요청 시간을 기초로 송신할 스트림을 선택하여 클라이언트에 송신하는 단계; 를 포함한다. 스트림은 데이터 전송시 응용 계층이 송신하고자 하는 데이터의 연속된 흐름인 것이고, 스트림 대기열에 배치된 스트림은 요청 시간이 빠른 것부터 순서대로 정렬되고, 측정된 데이터 크기가 작은 것부터 순서대로 정렬되어, 정렬된 순서대로 클라이언트에 송신되는 것이다.

대표도 - 도1



(52) CPC특허분류

H04L 47/628 (2013.01)

H04L 69/326 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711092830
과제번호	2017M3C4A7083676
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	차세대정보·컴퓨팅기술개발사업
연구과제명	Web 응용을 위한 Seamless 자율 전송 네트워킹 제어 프레임워크
기 여 율	1/1
과제수행기관명	고려대학교 산학협력단
연구기간	2019.03.01 ~ 2019.12.31

명세서

청구범위

청구항 1

서버에 의해 수행되는 스트림 스케줄링 방법에 있어서,

(a) 클라이언트로부터 데이터 전송 요청이 수신되면, 기설정된 조건에 따라 데이터를 복수의 스트림에 할당하는 단계;

(b) 상기 스트림에 할당된 데이터의 크기와 요청 시간을 측정하는 단계; 및

(c) 스트림 대기열에 상기 스트림을 배치하고, 측정된 상기 데이터 크기 및 요청 시간을 기초로 송신할 스트림을 선택하여 상기 클라이언트에 송신하는 단계; 를 포함하고,

상기 스트림은 데이터 전송시 응용 계층이 송신하고자 하는 데이터의 연속된 흐름인 것이고,

상기 (c) 단계에서 스트림 대기열에 배치된 스트림은 상기 요청 시간이 빠른 것부터 순서대로 정렬되고, 상기 측정된 데이터 크기가 작은 것부터 순서대로 정렬되어, 정렬된 순서대로 상기 클라이언트에 송신되는 것인,

스트림 스케줄링 방법.

청구항 2

제 1 항에 있어서,

상기 (a) 내지 (c) 단계는, 스트림 다중화를 지원하는 OSI 7 계층 모델의 전송 계층 프로토콜을 통해 수행되는 것이고,

상기 스트림 다중화는 서로 다른 스트림이 하나의 연결을 통해 전송되는 기법인 것인,

스트림 스케줄링 방법.

청구항 3

제 2 항에 있어서,

상기 전송 계층 프로토콜은 QUIC 또는 SCTP 중 어느 하나인 것인,

스트림 스케줄링 방법.

청구항 4

제 1 항에 있어서,

상기 (a) 단계에서 클라이언트로부터 복수의 데이터 전송 요청이 수신되고,

상기 (b) 단계에서 전송 요청이 수신된 복수의 데이터 각각의 크기와 요청 시간을 측정하는 것인,

스트림 스케줄링 방법.

청구항 5

제 1 항에 있어서,

상기 (a) 단계에서 상기 데이터 전송 요청은 클라이언트의 단일 응용에 의한 요청이고,

상기 (c) 단계에서 상기 송신할 스트림은 상기 클라이언트의 단일 응용에 송신되는 것인,

스트림 스케줄링 방법.

청구항 6

제 4 항에 있어서,

상기 (a) 단계에서 복수의 데이터 전송 요청이 수신되면, 데이터간에 의존성이 있거나 데이터의 전송 순서가 지켜져야 하는 데이터는 동일 스트림에 할당하고, 서로 독립되어 관련이 없는 데이터는 서로 다른 스트림에 할당하는 것인,

스트림 스케줄링 방법.

청구항 7

스트림 스케줄링 서버에 있어서,

스트림 스케줄링 프로그램이 저장된 메모리;

상기 메모리에 저장된 프로그램을 실행하는 프로세서를 포함하며,

상기 프로세서는 상기 프로그램의 실행에 의해, 클라이언트로부터 데이터 전송 요청이 수신되면, 기설정된 조건에 따라 데이터를 복수의 스트림에 할당하고, 상기 스트림에 할당된 데이터의 크기와 요청 시간을 측정하고, 스트림 대기열에 상기 스트림을 배치하고, 스트림 대기열에 배치된 스트림은 상기 요청 시간이 빠른 것부터 순서대로 정렬되고, 상기 측정된 데이터 크기가 작은 것부터 순서대로 정렬되어, 측정된 상기 데이터 크기 및 요청 시간을 기초로 정렬된 순서대로 송신할 스트림을 선택하여 상기 클라이언트에 송신하고,

상기 스트림은 데이터 전송시 응용 계층이 송신하고자 하는 데이터의 연속된 흐름인 것인,

스트림 스케줄링 서버.

청구항 8

제 7 항에 있어서,

상기 프로세서는 스트림 다중화를 지원하는 OSI(Open System Interconnection) 7 계층 모델의 전송 계층 프로토콜을 통해 스트림 스케줄링을 수행하는 것이고,

상기 스트림 다중화는 서로 다른 스트림이 하나의 연결을 통해 전송되는 기법인 것인,

스트림 스케줄링 서버.

청구항 9

제 8 항에 있어서,

상기 전송 계층 프로토콜은 QUIC 또는 SCTP 중 어느 하나인 것인,

스트림 스케줄링 서버.

청구항 10

제 7 항에 있어서,

상기 프로세서는 클라이언트로부터 복수의 데이터 전송 요청을 수신하고, 상기 전송 요청이 수신된 복수의 데이터 각각의 크기와 요청 시간을 측정하는 것인,

스트림 스케줄링 서버.

청구항 11

제 7 항에 있어서,

상기 데이터 전송 요청은 클라이언트의 단일 응용에 의한 요청이고,

상기 송신할 스트림은 상기 클라이언트의 단일 응용에 송신되는 것인,

스트림 스케줄링 서버.

청구항 12

제 10 항에 있어서,

상기 프로세서는 클라이언트로부터 복수의 데이터 전송 요청이 수신되면, 데이터간에 의존성이 있거나 데이터의 전송 순서가 지켜져야 하는 데이터는 동일 스트림에 할당하고, 서로 독립되어 관련이 없는 데이터는 서로 다른 스트림에 할당하는 것인,

스트림 스케줄링 서버.

청구항 13

제 1 항에 따르는 스트림 스케줄링 방법을 수행하기 위한 컴퓨터 프로그램이 기록된 비일시적 컴퓨터 판독가능 기록매체.

발명의 설명

기술 분야

[0001] 본 발명은 스트림 스케줄링 방법 및 서버에 관한 것으로, 보다 상세하게는 OSI(Open System Interconnection) 7 계층 모델에서의 전송 계층 프로토콜이 스트림 다중화(stream multiplexing)를 지원할 때 어떤 스트림의 데이터를 언제, 얼마만큼 송신할 것인지를 결정하는 스트림 스케줄링 방법에 관한 것이다.

배경 기술

[0002] OSI 7 계층 모델의 전송 계층은 양 종단 간의 신뢰성 있는 데이터 전송을 담당하며 이 계층의 대표적인 프로토콜인 TCP의 경우 전송에 실패한 패킷을 재전송하는 방식으로 신뢰성 있는 통신을 제공한다. 그러나 하나의 TCP 연결을 이용하여 서로 독립적인 다수의 데이터를 보내려 하는 경우 하나의 바이트 스트림만 제공하는 TCP의 특성상 한 데이터의 전송이 다른 데이터 전송을 지연시키는 HOL 블로킹(HoLB : Head-of-Line Blocking) 문제가 발생할 수 있다. 이 문제를 해결하기 위해 전송 단위로 다수의 스트림을 지원하고 하나의 연결 내에서 이 스트림들을 동시에 전송하는 스트림 다중화 기능을 지원하는 전송 계층 프로토콜이 제안되었다. 대표적으로 QUIC, SCTP 두 프로토콜이 여기에 해당한다.

[0003] 전송 계층 프로토콜의 스트림이란 서로 독립적인 데이터를 전송하기 위한 하나의 단위이다. 한 스트림 내부에서 전송되는 데이터들은 전송 순서대로 수신이 완료되어야 하지만 어떤 스트림과 다른 스트림 데이터의 송신 순서와 시간은 정해진 바가 없다. 따라서 다수의 스트림이 한 연결 내에 존재하는 상황에서 어떤 스트림을 언제, 얼마만큼 보낼 것인지를 결정하는 스트림 스케줄링 문제가 발생한다.

[0004] 패킷 스위치 네트워크와 같이 여러 주체가 공유된 자원을 나누어 사용하는 경우 이들 간의 공평한 자원 사용이 일반적인 고려 사항이 된다. 따라서 자원을 할당한 결과 개별 주체가 대역폭, 지연 시간 측면에서 공평한 서비스를 받았는지를 측정하여 자원 할당 방법이 공평성이 있는지를 확인하게 된다. 특정 주체가 악의적으로 자원을 많이 사용하여 다른 주체가 더 적은 자원을 할당받게 되는 문제를 막기 위한 방법도 필요하다. Fair Queuing 방법과 그 변형은 여러 주체가 네트워크 자원을 나누어 사용하려 하는 상황에서 대역폭과 지연 시간 공평성 측면에서 이점이 있다고 알려져 있어 흔히 사용되는 방법이다.

[0005] 하지만 스트림 다중화가 지원되는 전송 계층 프로토콜에서의 스트림 스케줄링 문제는 일반적인 네트워크에서의 자원 할당 문제와는 다르다. 일반적인 네트워크 자원 할당 문제는 개별 주체가 공유하는 자원을 더 많이 할당받고자 한다고 가정한다. 그러나 전송 계층 프로토콜의 개별 스트림은 본래 하나의 응용이 사용하는 것으로, 더 많은 자원 할당을 위해 스트림 간의 경쟁이 일어나지 않는다. 성능 지표 역시 개별 스트림별로 측정하는 것이 아니라 이들 스트림을 사용하는 상위 계층인 응용 수준에서 측정된다.

선행기술문헌

특허문헌

[0006] (특허문헌 0001) 대한민국 공개특허공보 제10-2006-0096077호 (발명의 명칭: 다중 채널들을 통해 데이터 통신하기 위한 방법 및 장치)

발명의 내용

해결하려는 과제

- [0007] 본 발명은 전술한 문제점을 해결하기 위한 것으로, 응용이 스트림 다중화가 지원되는 전송 계층 프로토콜을 이용하여 다수의 독립된 데이터를 전송할 때 개별 데이터 전송의 지연 시간을 줄이기 위해, 각 데이터 전송을 위한 스트림들이 송신 대기중인 상황에서 어떤 스트림을 보낼지를 결정하는 스트림 스케줄링 방법을 제공하는 것을 기술적 과제로 한다.
- [0008] 모든 스트림이 서로 다른 주체가 아닌 하나의 주체인 상위 응용이 이용하는 전송 단위라는 점에 기초하여, 스트림 간의 공평성을 우선순위로 두는 것이 아니라 먼저 전송이 완료될 것이 예상되는 스트림부터 송신하는 원칙을 바탕으로 하는 스트림 스케줄링 방법을 제공한다.
- [0009] 다만, 본 실시예가 이루고자 하는 기술적 과제는 상기된 바와 같은 기술적 과제로 한정되지 않으며, 또 다른 기술적 과제들이 존재할 수 있다.

과제의 해결 수단

- [0010] 상술한 기술적 과제를 해결하기 위한 기술적 수단으로서, 본 개시의 제 1측면에 따른 서버에 의해 수행되는 스트림 스케줄링 방법은, 클라이언트로부터 데이터 전송 요청이 수신되면, 기설정된 조건에 따라 데이터를 복수의 스트림에 할당하는 단계; 스트림에 할당된 데이터의 크기와 요청 시간을 측정하는 단계; 및 스트림 대기열에 스트림을 배치하고, 측정된 데이터 크기 및 요청 시간을 기초로 송신할 스트림을 선택하여 클라이언트에 송신하는 단계;를 포함한다. 스트림은 데이터 전송시 응용 계층이 송신하고자 하는 데이터의 연속된 흐름인 것이고, 스트림 대기열에 배치된 스트림은 요청 시간이 빠른 것부터 순서대로 정렬되고, 측정된 데이터 크기가 작은 것부터 순서대로 정렬되어, 정렬된 순서대로 클라이언트에 송신되는 것이다.
- [0011] 상술한 기술적 과제를 해결하기 위한 기술적 수단으로서, 본 개시의 제 2측면에 따른 서버에 의해 수행되는 스트림 스케줄링 서버는, 스트림 스케줄링 프로그램이 저장된 메모리; 상기 메모리에 저장된 프로그램을 실행하는 프로세서를 포함한다. 상기 프로세서는 상기 프로그램의 실행에 의해, 클라이언트로부터 데이터 전송 요청이 수신되면, 기설정된 조건에 따라 데이터를 복수의 스트림에 할당하고, 상기 스트림에 할당된 데이터의 크기와 요청 시간을 측정하고, 스트림 대기열에 상기 스트림을 배치하고, 스트림 대기열에 배치된 스트림은 상기 요청 시간이 빠른 것부터 순서대로 정렬되고, 상기 측정된 데이터 크기가 작은 것부터 순서대로 정렬되어, 측정된 상기 데이터 크기 및 요청 시간을 기초로 정렬된 순서대로 송신할 스트림을 선택하여 상기 클라이언트에 송신한다. 상기 스트림은 데이터 전송시 응용 계층이 송신하고자 하는 데이터의 연속된 흐름인 것이다.

발명의 효과

- [0012] 전술한 본원의 과제 해결 수단 중 어느 하나에 의하면, 개별 데이터들의 전송 완료 시간이 줄어들어 따라 이를 수신하는 응용이 먼저 도착한 데이터를 신속히 활용하거나 처리할 수 있다. 마지막 데이터의 전송이 끝나는 시간은 어떠한 스트림 스케줄링 방법을 사용하더라도 차이가 없지만, 마지막 이외의 데이터의 경우 본 발명을 통해 전송 완료 시간이 감소하게 되며 따라서 전체 데이터 활용의 효율성은 향상될 수 있다.

도면의 간단한 설명

- [0013] 도 1은 본 발명의 일 실시예에 따른 스트림 스케줄링 시스템의 구성을 도시한 블록도이다.
- 도 2는 본 발명의 일 실시예에 따른 스트림 스케줄링 서버의 구성을 도시한 블록도이다.
- 도 3은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법을 설명하는 도면이다.
- 도 4는 본 발명의 일 실시예에 따른 스트림 스케줄링 방법을 설명하는 순서도이다.
- 도 5는 본 발명의 일 실시예에 따른 스트림 스케줄링 방법에서 스트림 전송시 지연을 설명하는 도면이다.
- 도 6은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 적용을 설명하는 도면이다.
- 도 7은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 효과를 설명하는 도면이다.
- 도 8은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 효과를 설명하기 위해 선입선처리(FCFS: First-

Come, First-Served) 기법의 효과를 설명하는 도면이다.

도 9는 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 효과를 설명하기 위해 라운드 로빈(Round Robin)기법의 효과를 설명하는 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0014] 아래에서는 첨부한 도면을 참조하여 본원이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본원의 실시예를 상세히 설명한다. 그러나 본원은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본원을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.
- [0015] 본원 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다.
- [0016] 본원 명세서 전체에서, 어떤 부재가 다른 부재 “상에” 위치하고 있다고 할 때, 이는 어떤 부재가 다른 부재에 접해 있는 경우뿐 아니라 두 부재 사이에 또 다른 부재가 존재하는 경우도 포함한다.
- [0017] 이하 첨부된 도면을 참고하여 본 발명의 일 실시예를 상세히 설명하기로 한다.
- [0018] OSI 7 계층은 네트워크에서 통신이 일어나는 과정을 7단계로 나눈 것으로, 통신이 일어나는 과정을 단계적으로 파악하여 흐름을 한눈에 알아보고 관리하기 위한 모델이다. OSI 7 계층은 물리 계층(Physical Layer), 데이터링크 계층, 네트워크 계층, 전송 계층, 세션 계층, 표현 계층, 응용 계층으로 구성된다.
- [0019] 응용 계층(Application layer)은 응용 프로세스와 직접 관계하여 일반적인 응용 서비스를 수행한다. 일반적인 응용 서비스는 관련된 응용 프로세스들 사이의 전환을 제공한다.
- [0020] 전송 계층(Transport layer)은 양 끝단(End to end)의 사용자들이 신뢰성 있는 데이터를 주고 받을 수 있도록 해 주어, 상위 계층들이 데이터 전달의 유효성이나 효율성을 생각하지 않도록 해준다. 시퀀스 넘버 기반의 오류 제어 방식을 사용한다. 전송 계층은 특정 연결의 유효성을 제어하고, 일부 프로토콜은 상태 개념이 있고 (stateful), 연결 기반(connection oriented)이다. 이는 전송 계층이 패킷들의 전송이 유효한지 확인하고 전송 실패한 패킷들을 다시 전송한다는 것을 뜻한다. 또한 전송 계층은 종단간(end-to-end) 통신을 다루는 최하위 계층으로 종단간 신뢰성 있고 효율적인 데이터를 전송하며, 기능은 오류검출 및 복구와 흐름제어, 중복검사 등을 수행한다.
- [0021] 가장 잘 알려진 전송 계층의 예는 TCP이며, 스트림 다중화를 지원하는 전송 계층 프로토콜은 QUIC 또는 SCTP이 있다. 스트림 다중화는 하나의 연결 내에서 전송 단위로 다수의 스트림을 지원하여 다수의 스트림들을 동시에 전송하는 기능이다. 이 때 스트림은 응용 계층이 송신하고자 하는 바이트의 연속된 흐름을 추상화 한 것이다.
- [0022] 스트림 제어 전송 프로토콜(SCTP: Stream Control Transmission Protocol)은 컴퓨터 네트워킹에서 전송 계층 프로토콜의 하나로써, 잘 알려진 프로토콜인 전송 제어 프로토콜(TCP), 사용자 데이터그램 프로토콜(UDP)와 비슷한 역할을 수행한다. TCP와 UDP의 동일한 서비스 기능들 가운데 일부를 제공한다. TCP처럼 연결지향적 프로토콜이며 혼잡 제어를 통해 신뢰성 있는 순차적 메시지 전송을 보장한다.
- [0023] QUIC (Quick UDP Internet Connections) 프로토콜은 지연시간이 적은 인터넷 프로토콜로 계층 추가를 통해 TCP와 같은 신뢰성을 제공하고, 하나의 연결로 다수의 병렬 스트림으로 데이터를 동시 전송하며, 순서에 맞는 전송 처리를 하는 특징이 있다.
- [0024] 도 1은 본 발명의 일 실시예에 따른 스트림 스케줄링 시스템의 구성을 도시한 블록도이다.
- [0025] 스트림 스케줄링 방법은 서버(100) 또는 클라이언트(200)에서 실시될 수 있다. 서버(100)는 클라이언트(200)의 요청에 따라 데이터를 클라이언트(200)에 전송한다. 클라이언트(200)는 서버(100)에 보내는 한 번의 요청에 여러 데이터를 요청하며, 이 데이터들은 서로 독립된 데이터로써 서로 다른 스트림을 통해 전송될 수 있다. 서버(100)는 데이터를 가지고 있는 입장이므로 각 데이터의 크기를 미리 알고 있을 수 있다. 또한 클라이언트(200)가 서버(100)에 데이터를 요청하는 횟수에는 제한이 없다.
- [0026] 도 2는 본 발명의 일 실시예에 따른 스트림 스케줄링 서버의 구성을 도시한 블록도이다.
- [0027] 도시된 바와 같이 스트림 스케줄링 서버(100)는 통신 모듈(110), 메모리(120), 프로세서(130), 데이터베이스

(140) 및 입력모듈(150)을 포함할 수 있다.

- [0028] 통신모듈(110)은 접속된 클라이언트(200)과 데이터를 송수신할 수 있다. 통신모듈(110)은 다른 네트워크 장치와 유무선 연결을 통해 제어 신호 또는 데이터 신호와 같은 신호를 송수신하기 위해 필요한 하드웨어 및 소프트웨어를 포함하는 장치일 수 있다.
- [0029] 메모리(120)에는 스트림 스케줄링 프로그램이 저장된다. 스트림 스케줄링 프로그램은 클라이언트로부터 데이터 전송 요청이 수신되면, 기설정된 조건에 따라 데이터를 스트림에 할당하고, 상기 스트림에 할당된 데이터의 크기와 요청 시간을 측정하고, 스트림 대기열에 상기 스트림을 배치하고, 측정된 상기 데이터 크기 및 요청 시간을 기초로 송신할 스트림을 선택하여 클라이언트에 송신한다.
- [0030] 이러한 메모리(120)에는 스트림 스케줄링 서버(100)의 구동을 위한 운영 체제나 스트림 스케줄링 프로그램의 실행 과정에서 발생하는 여러 종류가 데이터가 저장된다.
- [0031] 이때, 메모리(120)는 전원이 공급되지 않아도 저장된 정보를 계속 유지하는 비휘발성 저장장치 및 저장된 정보를 유지하기 위하여 전력이 필요한 휘발성 저장장치를 통칭하는 것이다.
- [0032] 또한, 메모리(120)는 프로세서(130)가 처리하는 데이터를 일시적 또는 영구적으로 저장하는 기능을 수행할 수 있다. 여기서, 메모리(120)는 저장된 정보를 유지하기 위하여 전력이 필요한 휘발성 저장장치 외에 자기 저장 매체(magnetic storage media) 또는 플래시 저장 매체(flash storage media)를 포함할 수 있으나, 본 발명의 범위가 이에 한정되는 것은 아니다.
- [0033] 프로세서(130)는 메모리(120)에 저장된 프로그램을 실행하되, 스트림 스케줄링 프로그램의 실행에 따르는 전체 과정을 제어한다. 프로세서(130)가 수행하는 각각의 동작에 대해서는 추후 보다 상세히 살펴보기로 한다.
- [0034] 이러한 프로세서(130)는 데이터를 처리할 수 있는 모든 종류의 장치를 포함할 수 있다. 예를 들어 프로그램 내에 포함된 코드 또는 명령으로 표현된 기능을 수행하기 위해 물리적으로 구조화된 회로를 갖는, 하드웨어에 내장된 데이터 처리 장치를 의미할 수 있다. 이와 같이 하드웨어에 내장된 데이터 처리 장치의 일 예로써, 마이크로프로세서(microprocessor), 중앙처리장치(central processing unit: CPU), 프로세서 코어(processor core), 멀티프로세서(multiprocessor), ASIC(application-specific integrated circuit), FPGA(field programmable gate array) 등의 처리 장치를 망라할 수 있으나, 본 발명의 범위가 이에 한정되는 것은 아니다.
- [0035] 데이터베이스(140)는 프로세서(130)의 제어에 따라, 스트림 스케줄링 시스템에 필요한 데이터를 저장 또는 제공한다. 이러한 데이터베이스(140)는 메모리(120)와는 별도의 구성 요소로서 포함되거나, 또는 메모리(120)의 일부 영역에 구축될 수도 있다.
- [0036] 도 3은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법을 설명하는 도면이다.
- [0037] 클라이언트에 의해 데이터가 요청되면, 스트림 할당부는 데이터의 독립성 여부에 따라 데이터를 특정 스트림에 배치할 수 있다. 서로 독립되어 관련이 없다고 판단되는 데이터는 서로 다른 스트림에 할당되며, 의존성이 있으며 데이터의 전송 순서가 지켜져야 하는 데이터는 같은 스트림에 할당된다. 하나의 메시지에 포함된 데이터들은 관련성이 있는 데이터로 판단될 수 있다. 이 때 메시지는 데이터의 논리적 단위로 전송 처리를 위한 제어 또는 변조 신호가 아닌, 어떤 의미 단위로 구분되고, 그 내부에 실제 정보가 들어있는 신호를 의미한다.
- [0038] 데이터 측정부는 요청된 데이터의 크기와 요청 시간을 측정하여 각 데이터에 할당한다. 클라이언트의 i 번째 요청에 대해, 요청이 이루어진 시간을 t_i , 요청된 데이터의 수를 n_i 라고 하면, 데이터 측정부는 i 번째 요청에 포함된 n_i 개의 데이터의 각 크기인 $s_1^i, s_2^i, \dots, s_{n_i}^i$ 을 계산할 수 있다. 한편, 각 요청이 이루어진 시간 t_i 는 자연스럽게 하기 수학적식을 따르게 된다.
- [0039] [수학적식 1]
- [0040] $t_1 < t_2 < \dots < t_{i-1} < t_i < t_{i+1} < \dots$
- [0041] 이를 바탕으로 데이터 측정부는 요청된 각 데이터에 대해 요청 시간과 데이터의 크기를 나타내는 값인 (t, s) 를 부여한다. 스트림 할당부와 데이터 측정부에 의해 스트림의 할당과 데이터의 크기가 측정된 데이터는 스트림 대기열에 배치된다. 그 이후 스케줄러는 스트림 대기열의 데이터에 대해 i 의 값이 작은 것부터 오름차순으로, 그

다음은 s 의 값이 작은 것부터 오름차순으로 전송할 데이터를 선택한다. 여러 데이터에 부여된 (t, s) 의 값이 같은 경우 그 중 하나를 임의로 선택할 수 있다.

- [0042] 패킷 손실이 감지되어 재전송이 일어나는 경우 재전송 패킷을 가장 먼저 송신해야 하므로 재전송이 일어난 경우는 고려하지 않는다. 또한 본 스트림 스케줄링의 대상이 되는 것은 응용 데이터를 전달하는 스트림에 대한 것으로, 연결 수립 및 관리를 위한 데이터 전송은 그 대상이 아니다. 송신할 데이터가 결정된 이후에는 패킷 데이터 대기열로 전달된다.
- [0043] 도 4는 본 발명의 일 실시예에 따른 스트림 스케줄링 방법을 설명하는 순서도이다.
- [0044] 서버(100)는 클라이언트(200)로부터 데이터 전송 요청이 수신되면(S110), 기설정된 조건에 따라 데이터를 스트림에 할당한다(S120). 클라이언트(200)는 복수의 데이터 전송을 요청할 수 있고, 서버(100)는 요청된 복수의 데이터에 대해 데이터간에 의존성이 있거나 데이터의 전송 순서가 지켜져야 하는 데이터는 동일 스트림에 할당하고, 서로 독립되어 관련이 없는 데이터는 서로 다른 스트림에 할당할 수 있다.
- [0045] 서버(100)는 스트림에 할당된 데이터의 크기와 요청 시간을 측정한다(S130).
- [0046] 서버(100)는 스트림 대기열에 스트림을 배치한다(S140). 스트림 대기열은 데이터가 할당된 스트림들이, 스트림 스케줄러에 의해 선택되기 전 대기하는 곳으로, 스트림 대기열에 저장된 순서대로 송신되는 것은 아니다.
- [0047] 측정된 데이터 크기 및 요청 시간을 기초로 송신할 스트림을 선택하여 클라이언트에 송신한다(S150).
- [0048] 서버(100)는 측정된 요청 시간이 작은 스트림부터 먼저 선택하여 클라이언트에 송신하거나, 측정된 데이터 크기가 작은 스트림부터 먼저 선택하여 클라이언트에 송신할 수 있다.
- [0049] 이 때, 스트림 스케줄링 방법은 스트림 다중화를 지원하는 OSI 7 계층 모델의 전송 계층 프로토콜을 통해 수행될 수 있다. 예를 들어 전송 계층 프로토콜은 QUIC 또는 SCTP 중 어느 하나인 것일 수 있다. 또한 본 발명에서 수행되는 스트림 스케줄링 방법은 클라이언트의 단일 응용에서 요청한 데이터 및 스트림을 대상으로 하는 것일 수 있다.
- [0050] 도 5는 본 발명의 일 실시예에 따른 스트림 스케줄링 방법에서 스트림 전송시 지연을 설명하는 도면이다.
- [0051] 본 발명은 저지연 응용 서비스를 제공하기 위해 스트림 스케줄링 방법을 제공한다. 개별 스트림이 크기가 정해진 메시지를 보낼 때, 메시지가 전송 계층으로 요청된 시간부터 통신 상대방에게 전송이 완료되기까지의 시간을 지연(latency)이라 하고, 모든 메시지의 지연의 총합을 줄이는 것이 본 발명의 목표이다. 도5에 표시된 바와 같이 스트림별로 할당된 데이터의 크기가 다르기 때문에, 스트림을 전송하는 순서에 따라 각 스트림별 지연의 총합은 달라질 수 있다.
- [0052] 도 6은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 적용을 설명하는 도면이다.
- [0053] 본 발명의 효과를 확인하기 위해 본발명의 스트림 스케줄링, 선입선처리 스케줄링, 라운드로빈 스케줄링 각각을 사용하여 데이터를 송신한다. 매 송신은 라운드 단위로 이루어지며, 각 라운드에서 보내는 데이터 크기는 10, 20, 40, 40, 40으로 설정될 수 있다.
- [0054] 스트림의 크기는 A: 16seg, B: 9seg, C: 14seg, D: 4seg, E: 31seg, F: 11 seg, G: 15 seg, H: 18 seg, I: 3seg, J: 7 seg, K: 11seg 이다.
- [0055] 스트림 A,B는 라운드 1 전에 요청되고, 스트림 C,D는 라운드 2전에 요청되고, 스트림 E,F,G는 라운드3전에 요청되고, 스트림 H,I,J는 라운드4전에 요청되고, 스트림 K는 라운드 5전에 요청된다.
- [0056] 라운드당 송신 데이터 크기의 변화는 TCP, QUIC 등의 실제 사용되는 혼잡 제어 기법(AIMD 기반)과 유사하다. 매 전송에 걸리는 시간은 100 ms로 가정할 수 있다. 라운드가 시작되기 전에 송신 요청된 데이터만 그 라운드의 스케줄링 대상에 포함될 수 있다. 예를 들어 도 6의 스트림 C, D의 데이터는 라운드 1에 전송될 수 없고 라운드2에 스케줄링 될 것이다.
- [0057] 본 발명에 해당하는 Low Latency(LL) 스케줄링은 데이터의 크기가 작은 순서대로 전송한다. 본 발명의 결과는 도 7에서 설명한다.
- [0058] 선입선처리 스케줄링은 요청 시간 순서대로 전송하며, 요청 시간이 동일한 경우 임의의 순서 혹은 스트림 식별 번호 순서대로 전송한다. 선입선처리 스케줄링에 따른 결과는 도8에서 설명한다.

- [0059] 라운드로빈 스케줄링은 모든 스트림을 일정 단위로 나누어 공평하게 돌아가며 전송한다. 라운드로빈 스케줄링에 따른 결과는 도 9에서 설명한다.
- [0060] 어떤 스케줄링 기법을 사용하는 마지막 데이터 전송이 끝나가는 시점은 변함이 없으나 본 발명을 적용하면 개별 데이터의 지연 시간을 줄이는 이점을 얻을 수 있음을 확인할 수 있다.
- [0061] 도 7은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 효과를 설명하는 도면이다.
- [0062] 라운드 1에서는 스트림 A 중 1 및 스트림 B중 9를 전송하고, 라운드 2에서는 A:15, C:1, D:4를 전송한다. 라운드 3에서는 C:13, E:1, F:11, G:15 라운드 4에서는 E:30, I:3, J:7 을 전송한다. 라운드 5에서는 H:18,K:11을 전송하고 모든 스트림의 전송이 완료된다.
- [0063] 이 때 각 스트림별 지연시간은 A: 200 ms, B: 100 ms, C: 200 ms, D: 100 ms, E: 200 ms, F: 100 ms, G: 100 ms, H: 100 ms, I: 100 ms, J: 100 ms, K: 100 ms이다. 스트림별 지연시간을 모두 합친 총 지연시간은 1400 ms 이다.
- [0064] 도 8은 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 효과를 설명하기 위해 선입선처리 기법의 효과를 설명하는 도면이다.
- [0065] 라운드 1에서는 스트림 A 중 10을 전송하고, 라운드 2에서는 A:6, B:9, C:5를 전송한다. 라운드 3에서는 C:9, D:4, E:27 라운드 4에서는 E:4, F:11, G:15, H:10을 전송한다. 라운드 5에서는 H:8, I:3, J:7,K11을 전송하고 모든 스트림의 전송이 완료된다.
- [0066] 이 때 각 스트림별 지연시간은 A: 200 ms, B: 100 ms, C: 200 ms, D: 100 ms, E: 200 ms, F: 100 ms, G: 100 ms, H: 200 ms, I: 100 ms, J: 100 ms, K: 100 ms이다. 스트림별 지연시간을 모두 합친 총 지연시간은 1500 ms 이다.
- [0067] 도 9는 본 발명의 일 실시예에 따른 스트림 스케줄링 방법의 효과를 설명하기 위해 라운드로빈 기법의 효과를 설명하는 도면이다.
- [0068] 라운드 1에서는 스트림 A :5 B:5를 전송하고, 라운드 2에서는 A:6, B:4, C:6, D:4를 전송한다. 라운드 3에서는 A:5, C:8, E:9, F:9, G:9 라운드 4에서는 E:11, F:2, G:6, H:11, I:3, J:7을 전송한다. 라운드 5에서는 E:11, H:7, K:11을 전송하고 모든 스트림의 전송이 완료된다.
- [0069] 이 때 각 스트림별 지연시간은 A: 300 ms, B: 200 ms, C: 200 ms, D: 100 ms, E: 300 ms, F: 200 ms, G: 200 ms, H: 200 ms, I: 100 ms, J: 100 ms, K: 100 ms이다. 스트림별 지연시간을 모두 합친 총 지연시간은 2000 ms 이다.
- [0070] 본 발명의 일 실시예는 컴퓨터에 의해 실행되는 프로그램 모듈과 같은 컴퓨터에 의해 실행가능한 명령어를 포함하는 기록 매체의 형태로도 구현될 수 있다. 컴퓨터 판독 가능 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 가용 매체일 수 있고, 휘발성 및 비휘발성 매체, 분리형 및 비분리형 매체를 모두 포함한다. 또한, 컴퓨터 판독 가능 매체는 컴퓨터 저장 매체를 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함한다.
- [0071] 본 발명의 방법 및 시스템은 특정 실시예와 관련하여 설명되었지만, 그것들의 구성 요소 또는 동작의 일부 또는 전부는 범용 하드웨어 아키텍처를 갖는 컴퓨터 시스템을 사용하여 구현될 수 있다.
- [0072] 전술한 본원의 설명은 예시를 위한 것이며, 본원이 속하는 기술분야의 통상의 지식을 가진 자는 본원의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.
- [0073] 본원의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본원의 범위에 포함되는 것으로 해석되어야 한다.

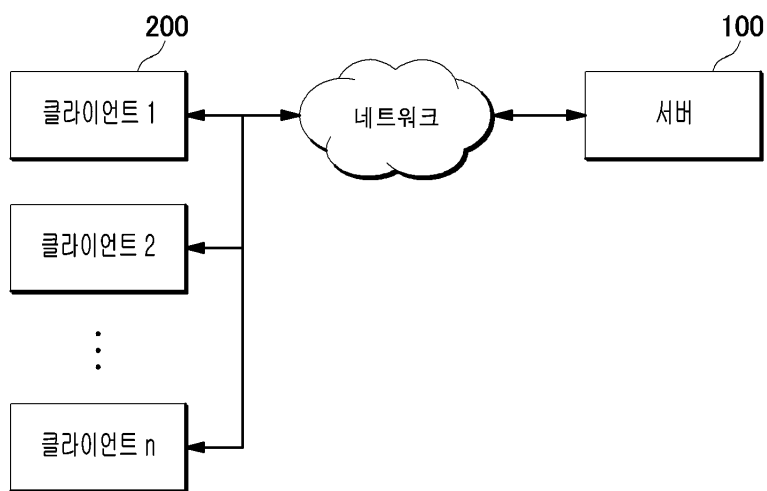
부호의 설명

[0074]

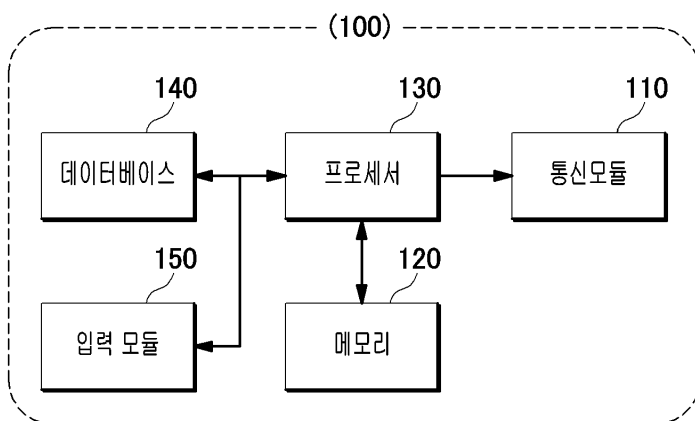
- 100: 서버
- 110: 통신 모듈
- 120: 메모리
- 130: 프로세서
- 140: 데이터베이스
- 150: 입력모듈
- 200: 클라이언트

도면

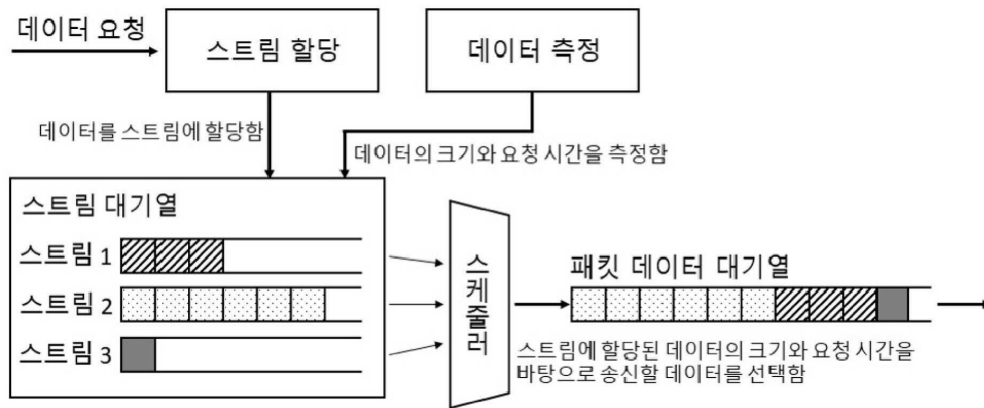
도면1



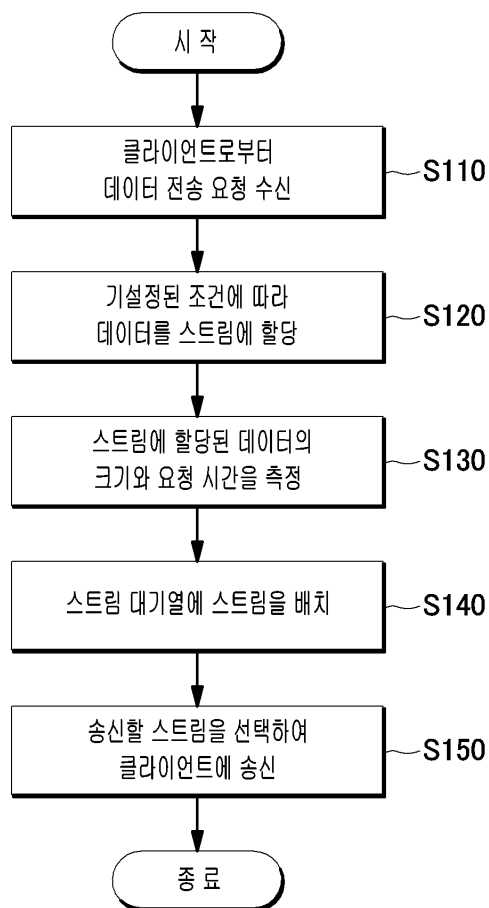
도면2



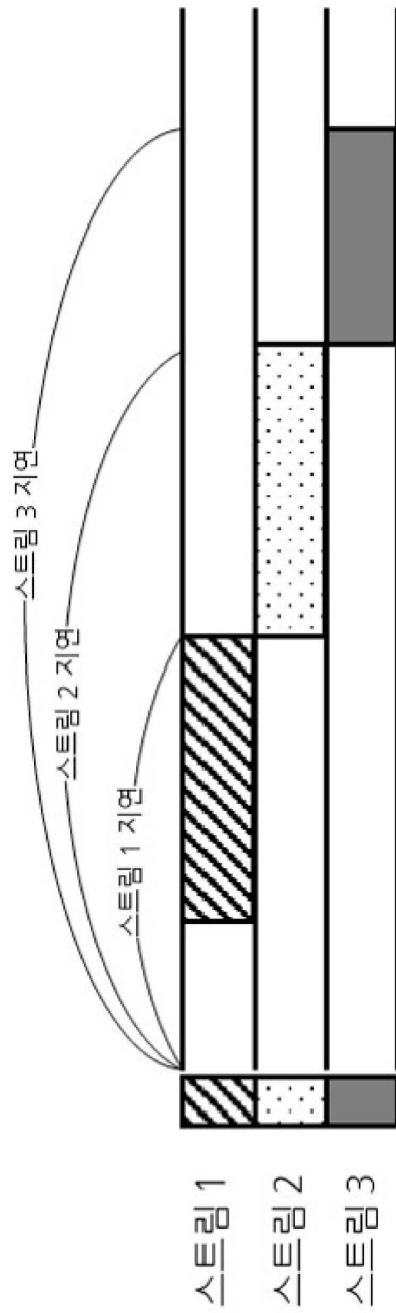
도면3



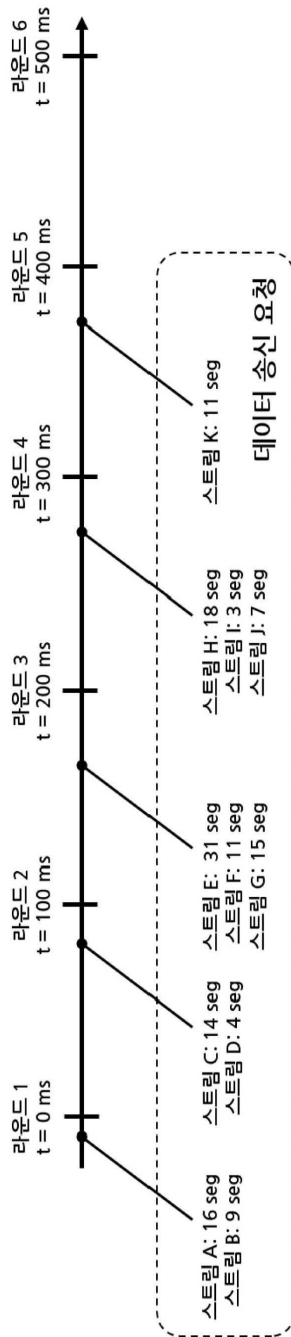
도면4



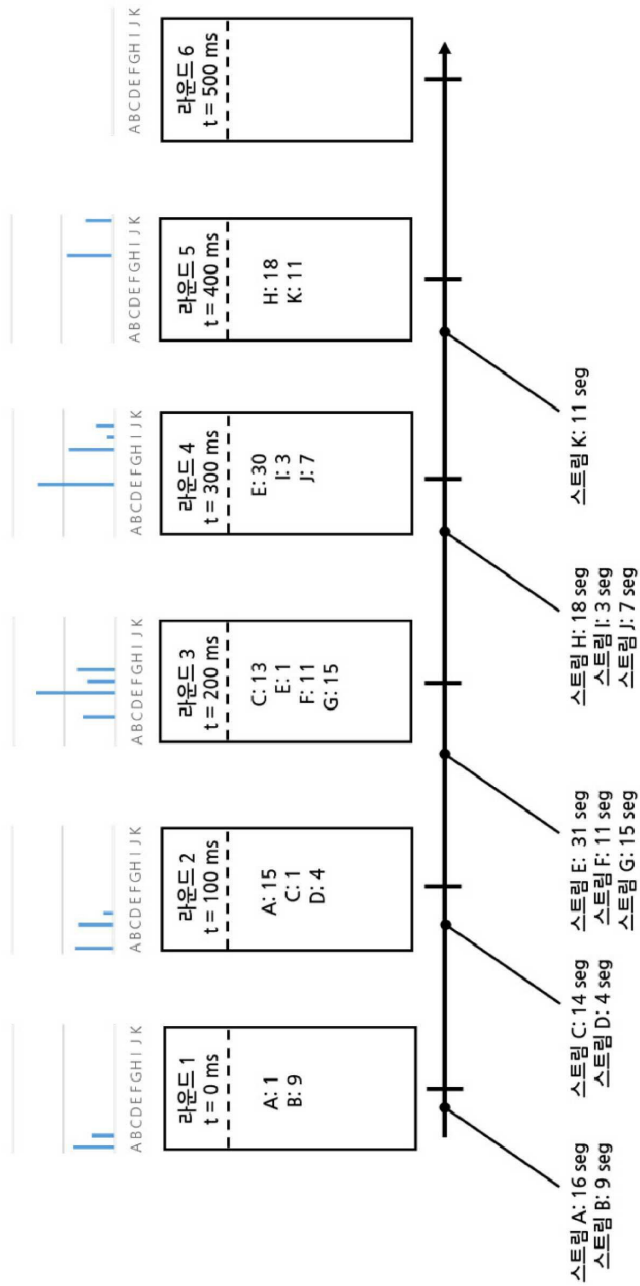
도면5



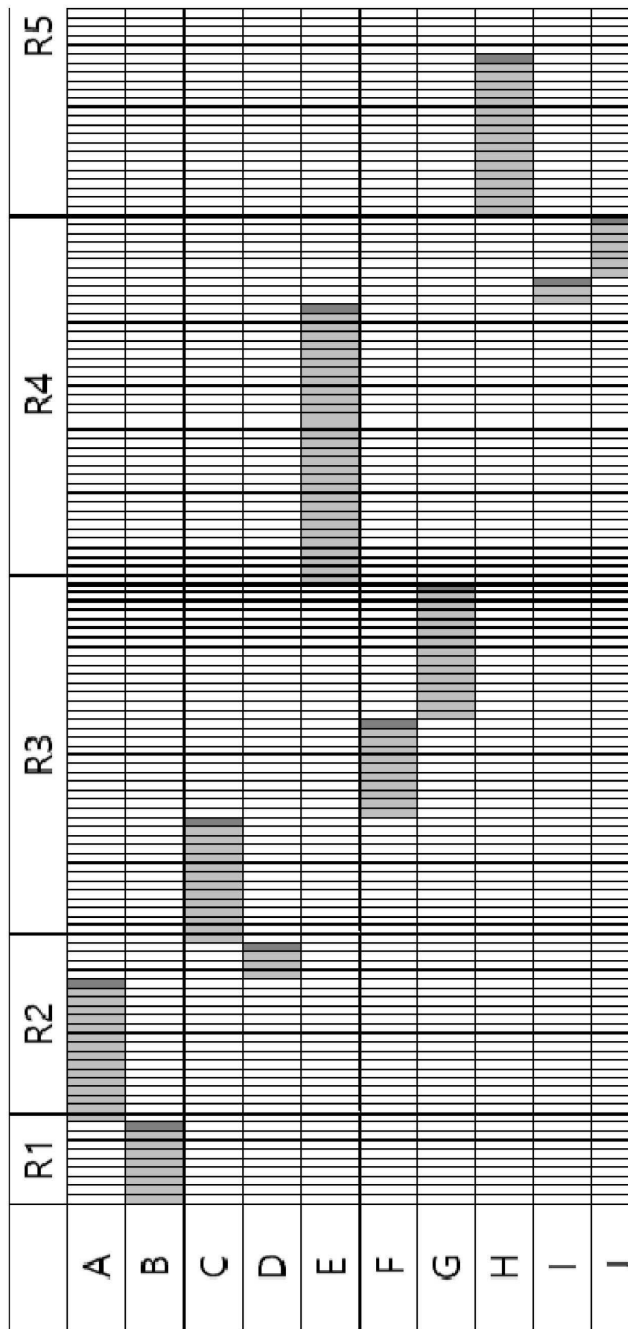
도면6



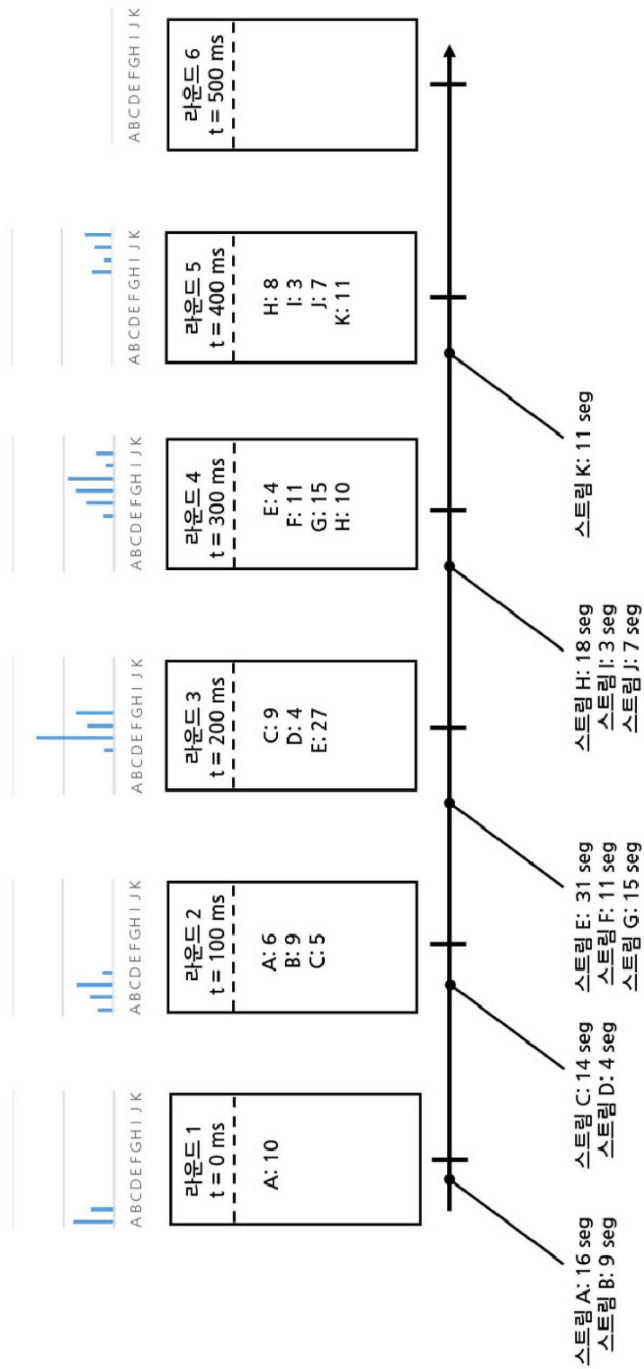
도면7a



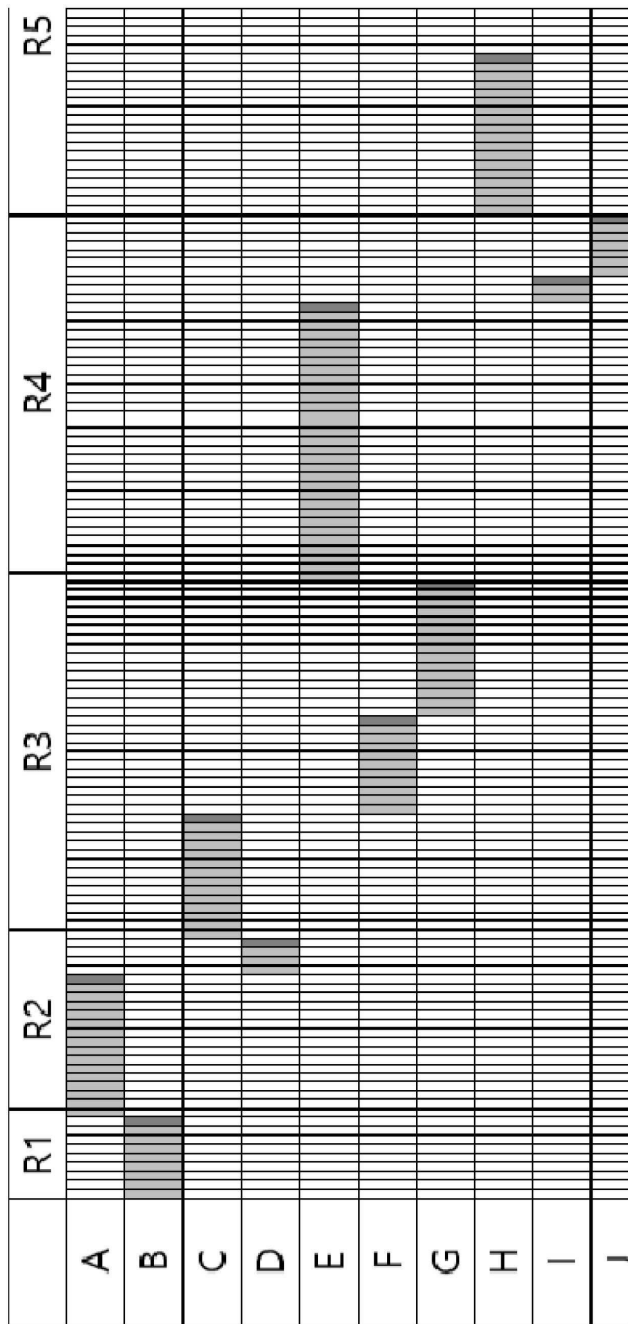
도면7b



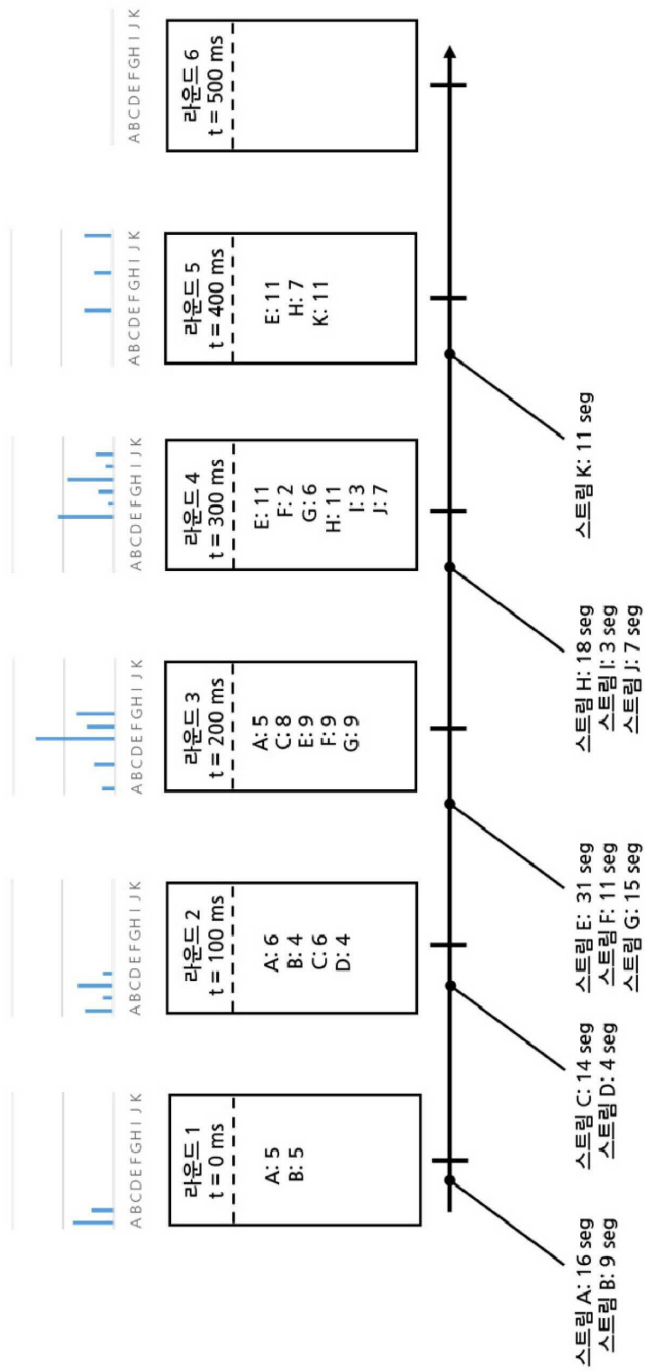
도면8a



도면8b



도면9a



도면9b

