

IP 방송제작 기술 5

PTP(Precision Time Protocol) 개론

글. 조인준 KBS 미디어기술연구소 차장

지난 연재에서 Ordinary Clock, Boundary Clock, Transparent Clock에 대해 알아보며 Ordinary Clock은 PTP 네트워크 동기화의 기준이 되는 Master Clock과 각종 비디오/오디오 신호를 처리하는 장비들인 Slave Clock을 지칭하고, Boundary Clock과 Transparent Clock은 네트워크 스위치들을 가리키며, 각 Clock은 어떤 특성을 갖는지를 말씀드렸습니다.

이번 연재에서는 Ordinary Clock, Boundary Clock, Transparent Clock들이 IP 네트워크에서 함께 동작하는 상황에서 나타나는 현상 및 이를 다루는 방법 등에 관해 설명하겠습니다.

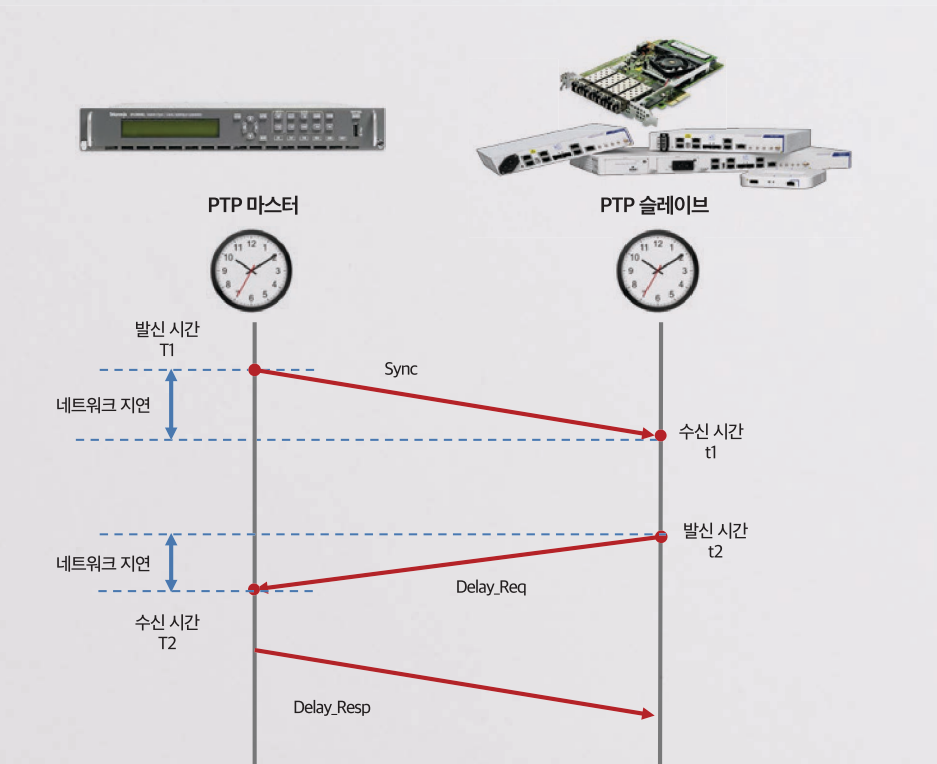


그림 1. Master Clock ↔ Slave Clock PTP 메시지 교환

지난 연재에서 설명해 드렸듯이 Master Clock과 Slave Clock 사이에는 Sync, Delay_Req(Delay Request), Delay_Resp(Delay Response)와 같은 PTP 메시지가 계속 오갑니다. 이 메시지들이 실제로 IP 네트워크에서 어떻게 이동하는지 설명하기 위해 우선 [그림 2]와 같은 PTP 장비 연결을 가정해보겠습니다. [그림 2]의 Master는 Master Clock, Slave는 Slave Clock을 나타냅니다. 앞선 연재에서 설명해 드렸듯이 [그림 2]의 Transparent Clock과 Boundary Clock은 모두 네트워크 스위치입니다. 차이점은 Transparent Clock은 Master Clock의 PTP 메시지에 해당 메시지가 자신을 통과하며 지연된 시간을 기록해준다는 것이고, Boundary Clock은 Master Clock에는 Slave 모드로 동작하여 동기화된 뒤에 자신에게 연결된 Slave Clock들에는 Master Clock처럼 동작하는 것입니다.

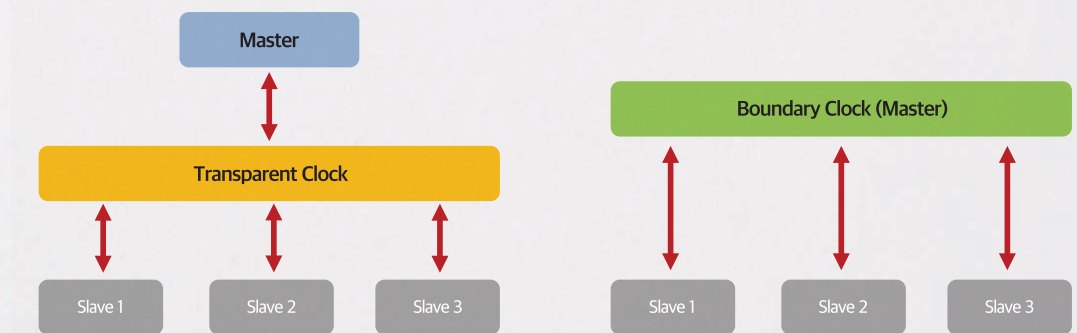


그림 2. PTP 동기화를 위한 Master Clock과 Slave Clock의 네트워크 연결

IP 네트워크에서 PTP 메시지의 흐름과 이로 인해 발생하는 현상들의 설명을 위해서 [그림 2] 좌측의 Transparent Clock의 경우를 통해 설명하겠습니다. PTP 메시지 흐름에 관해 Boundary Clock의 경우도 다르지 않으므로 불필요한 반복 없이 간결히 설명하기 위함입니다.

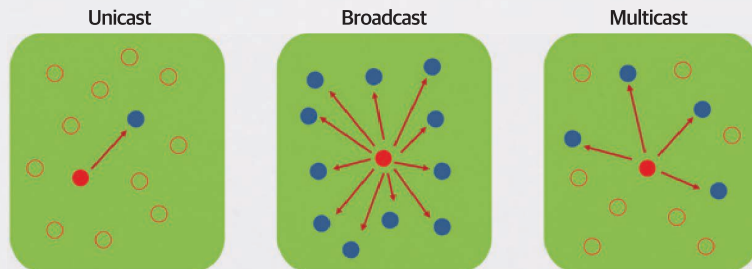


그림 3. Unicast, Broadcast, Multicast 전송방식

[그림 3]은 IP 네트워크에서 송신단과 수신단의 통신을 위한 데이터 전송 방식인 Unicast, Broadcast, Multicast를 나타냅니다. 각 방식에 대한 간략한 설명은 아래와 같습니다.

Unicast

- 네트워크에서 가장 많이 사용되는 방식으로 특정 수신장비에만 데이터 패킷을 전달하는 방식

Broadcast

- 네트워크에 연결되어 있는 모든 수신장비들에 동일한 데이터 패킷을 보내는 방식

Multicast

- 특정 그룹에 속한 수신장비들에 데이터 패킷을 전송하는 방식

그럼 Unicast, Broadcast, Multicast가 PTP 동기화와 어떤 관련이 있을까요? 일단 Broadcast 방식은 PTP 메시지 전송과 관련해 사용하기 힘든 방식이므로 이후의 설명에서 제외하도록 하겠습니다. Broadcast 방식이 왜 제외될 수밖에 없는지는 아래의 내용을 보시면 추가적 설명이 없이도 충분히 이해가 될 것입니다. Broadcast 방식을 제외한 Unicast, Multicast 방식이 PTP 메시지 전송과 어떤 관련이 있는지 설명을 위해 [그림 4]의 Master Clock이 보내는 Sync 메시지를 보겠습니다.

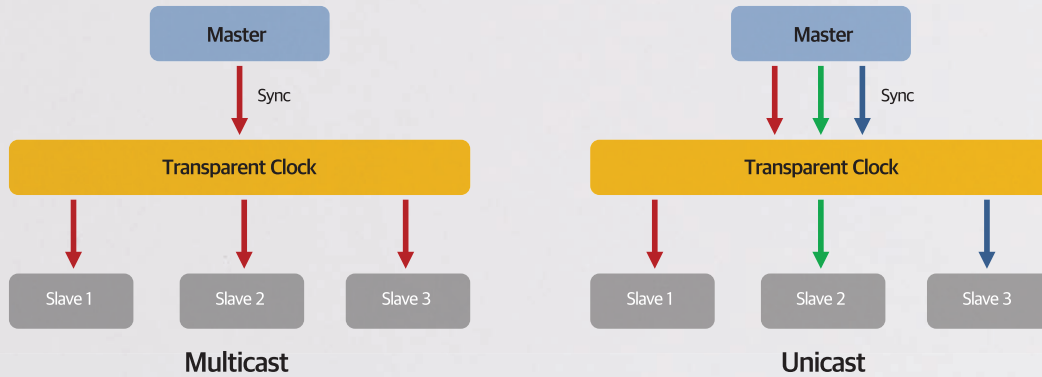


그림 4. Sync 메시지 전송 : Multicast vs Unicast

[그림 4] 좌측은 Multicast를 통해 Master Clock이 같은 그룹에 속해있는 Slave Clock들에 Sync 메시지를 보내는 경우입니다. 이 경우 Master Clock은 단 한 번의 Sync 메시지 발송만으로 Multicast 그룹으로 묶인 Slave Clock들에 Sync 메시지를 한 번에 모두 전달할 수 있습니다. [그림 4] 우측은 Unicast를 통해 Master Clock이 Sync 메시지를 보내는 경우입니다. Unicast로 보내는 탓에 각 Slave Clock마다 Sync 메시지를 다 따로 보내고 있습니다. 누가 보더라도 Master Clock이 Sync 메시지를 보낼 때는 Multicast로 보내는 것이 합리적입니다.

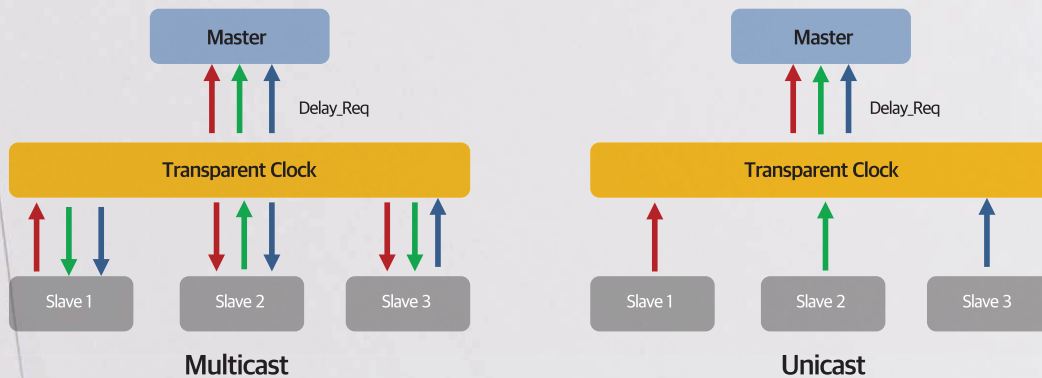


그림 5. Delay_Req 메시지 전송 : Multicast vs Unicast

Master Clock의 Sync 메시지를 받으면 각 Slave Clock들은 Delay_Req 메시지를 Master Clock에게 보냅니다. [그림 5]는 Multicast 상황에서 각 Slave Clock들이 Master Clock에게 Delay_Req 메시지를 보내고 있습니다. [그림 5]에서 빨간색 화살표는 'Slave 1', 초록색 화살표는 'Slave 2', 파란색 화살표는 'Slave 3'의 Delay_Req 메시지 흐름입니다. [그림 5] 좌측을 보면 'Slave 1'이 보낸 Delay_Req 메시지가 Master Clock에게 전달되는 것뿐만 아니라 같은 그룹에 속한 Slave 2와 Slave 3에도 전달됩니다. 이는 Slave 2와 Slave 3의 Delay_Req 메시지도 똑같이 해당됩니다. 반면 [그림 5]의 우측은 Unicast를 이용한 Delay_Req 메시지 전송입니다. 각 Slave Clock들의 Delay_Req 메시지는 Master Clock에게만 전달되고 다른 Slave Clock에게는 전달되지 않습니다. 그러므로 Delay_Req 메시지는 Unicast로 전송하는 것이 효율적입니다.

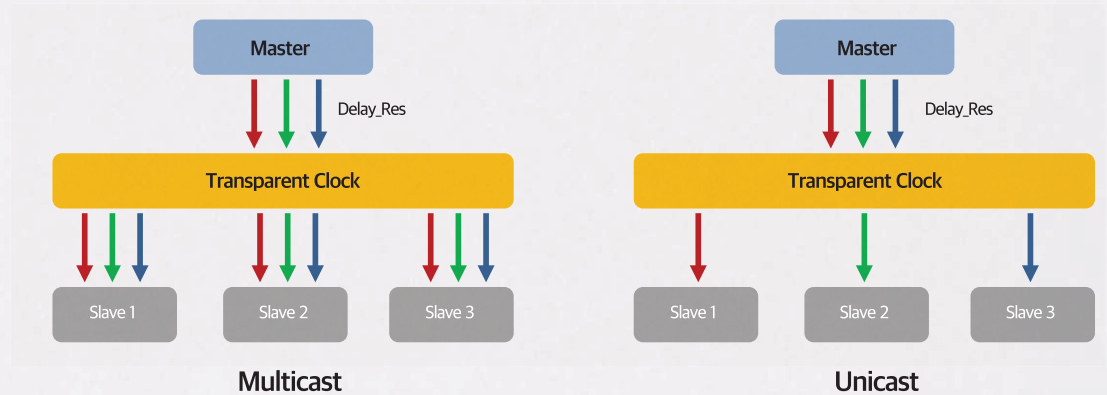


그림 6. Delay_Res 메시지 전송 : Multicast vs Unicast

Slave Clock의 Delay_Req 메시지를 받은 Master Clock은 Delay_Res 메시지를 해당 Slave Clock에게 전송합니다. [그림 6]은 Master Clock의 Delay_Res 메시지 흐름을 나타냅니다. [그림 6]도 [그림 5]와 마찬가지로 빨간색 화살표는 'Slave 1', 초록색 화살표는 'Slave 2', 파란색 화살표는 'Slave 3'의 Delay_Req 메시지 흐름입니다. Delay_Res 메시지도 Delay_Req 메시지와 마찬가지로 Multicast로 보낼 경우 각 Slave Clock이 받지 않아도 될 Delay_Res 메시지를 받게 됩니다. 반면 Unicast로 보내게 되면 각 Slave Clock이 자신이 받아야 할 Delay_Res 메시지만 수신하게 됩니다. Delay_Res 메시지의 경우도 Unicast를 사용하는 것이 효율적입니다.

이상 지금까지의 내용을 정리하면 다음과 같습니다. Master Clock이 자신과 연결된 Slave Clock들에 보내는 Sync 메시지는 Multicast로 전송하고, 각 Slave Clock들이 Master Clock에게 보내는 Delay_Req와 Delay_Res 메시지는 Unicast로 전송하는 하이브리드 방식이 PTP 동기화에 적합한 방식입니다.

지금까지 PTP 동기화 메시지의 네트워크 전송방식에 대해서 간단히 알아보았습니다. 다음 연재에서는 PTP 동기화가 된 방송장비 IP 네트워크가 어떻게 비디오/오디오의 Sync를 맞추는지를 알아보겠습니다. 📺