

CAN-over-Ethernet 을 위한 UDP 멀티캐스트

- 물리적 한계를 뛰어넘는 확장형 네트워크 구조-

1. 개요

본 문서에서는 UDP 멀티캐스트(UDP Multicast)를 사용하여 CAN 데이터를 이더넷 기반으로 효율적으로 터널링하는 방법을 설명합니다. 이를 통해 CAN 고유의 특성을 그대로 유지하면서도, 기존의 물리적·논리적 한계를 극복하는 방법을 다룹니다.

기존의 CAN 네트워크는 대규모 시스템으로 확장될 때 케이블 길이의 제약, 식별자(ID) 공간의 부족, 그리고 CPU 부하 증가 등으로 인해 빠르게 한계에 부딪히게 됩니다. UDP 멀티캐스트는 이러한 제약 조건을 극복할 수 있는 명확한 해법을 제공합니다. 본 기사에서는 CAN 버스의 근본적인 특성을 보존하면서도, CAN 데이터를 이더넷 위로 효율적으로 터널링하여 전송하는 구체적인 원리를 살펴봅니다.



2. 기존 CAN 네트워크의 한계

기존의 CAN CC(Classical CAN)는 견고하고 실시간성이 보장되는 통신을 제공하도록 설계되었지만, 물리적·논리적 제약에 묶여 있습니다. 통신 속도(비트 레이트)가 빨라질수록, 신호 전달 지연과 신호 무결성(Signal integrity) 문제로 인해 허용 가능한 최대 버스 길이는

급격히 줄어듭니다. 대표적으로 1Mbit/s 의 속도에서는 버스 길이가 보통 약 40m 로 제한됩니다. 수십 개의 CAN 노드(예: 64 개의 장치)가 연결되는 대규모 시스템 구성에서는 이러한 제약이 훨씬 더 심각해집니다. 단일 물리 버스 구조에서는 허용 길이를 초과할 뿐만 아니라, 여러 장치가 동시에 동일한 식별자(ID)를 전송할 경우 중재(Arbitration) 충돌이 발생하게 됩니다.

또 다른 핵심 과제는 제한된 식별자(ID) 공간입니다. CAN 버스에 연결된 모든 노드는 동일한 ID 네임스페이스를 공유합니다. 따라서 여러 장치가 완전히 동일한 CAN ID 를 사용하면 중재 충돌이 일어나거나 데이터가 잘못 해석되는 문제가 발생합니다. 이는 장치들이 고정되거나 하드코딩된 CAN ID 를 사용하는 환경에서 특히 치명적입니다. 그대로 둘 경우 ID 충돌이 불가피하기 때문에, 이러한 장치들을 하나의 공유 네트워크로 통합하는 것은 수정 없이는 불가능합니다.

여기에 더해, 네트워크 트래픽이 늘어날수록 시스템 부하도 함께 증가합니다. CAN 프레임이 수신될 때마다 인터럽트(Interrupt)가 발생하기 때문에, 메시지 전송 속도가 빨라지면 CPU 에 상당한 부담을 주게 됩니다.

Feature	CAN Standard	CAN-over-UDP Solution
Max Distance	~40m @ 1Mbps	Virtually unlimited (Fiber/Ethernet)
Node Count	Limited by physical loading	Scalable (64+ nodes)
ID Management	Must be unique on bus	Can be duplicated (mapped via IP)
CPU Load	High (Interrupt per frame)	Low (Frames bundled in UDP packets)

<표 1: CAN 표준과 CAN-over-UDP 비교>

3. 아키텍처적 접근 방식으로서의 CAN-over-Ethernet

이러한 한계점들을 극복하기 위해 CAN 데이터를 이더넷으로 전송할 수 있습니다. 개별 CAN 프레임들을 UDP/IP 패킷 내에 캡슐화(Encapsulation)하여 표준 이더넷 인프라를 통해 전송하는 방식입니다. 이를 통해 기존 물리 버스의 제약으로부터 완전히 자유로운 '가상 CAN 버스'가 만들어집니다.

실제 시스템 배포 환경에서는 각 CAN 노드가 전용 'CAN-to-Ethernet 게이트웨이'에

연결됩니다. 이 게이트웨이들은 이더넷 스위치를 통해 서로 연결되어 하나의 분산 시스템을 형성하며, 기존의 단일 공유 CAN 버스를 가상화되고 확장 가능한 백본망으로 효과적으로 대체합니다. 이러한 분리(Decoupling) 덕분에 한층 유연하고 분산된 시스템 설계가 가능해집니다.

4. 전송 프로토콜로서의 UDP

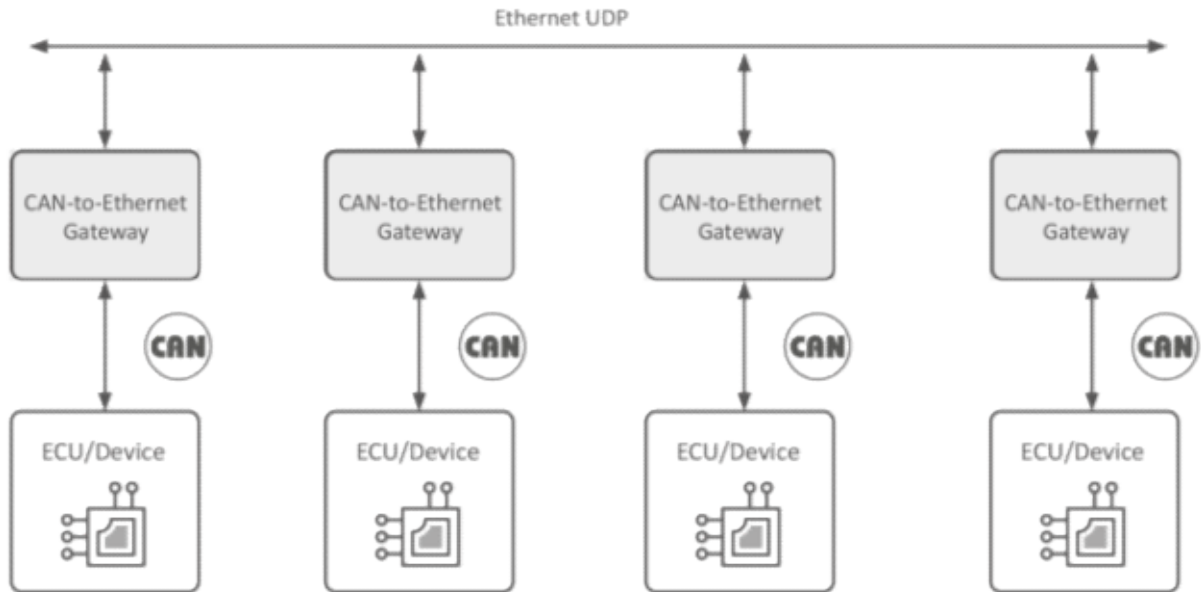
UDP는 의도적으로 선택된 전송 프로토콜입니다. TCP와 달리 UDP는 연결 지향형이 아니기 때문에 핸드셰이크(Handshake, 통신을 시작하기 전 두 장치가 서로 필요한 정보를 주고 받아 동기화, 협상을 하는 것), 흐름 제어(Flow control), 재전송(Retransmission) 같은 복잡한 매커니즘을 거치지 않습니다. 그 결과 지연 시간(Latency)이 단축되고 더욱 결정론적인(Deterministic) 동작이 가능해지는데, 이는 실시간성이 핵심인 CAN 통신에서 필수적인 요소입니다.

또 다른 장점은 처리 효율성입니다. 여러 개의 CAN 프레임을 단 하나의 UDP 패킷에 모아서(Aggregation) 보낼 수 있기 때문에, 수신측 시스템에서 발생하는 인터럽트 횟수를 대폭 줄여줍니다. 이를 통해 CPU 부하를 낮추고 전반적인 시스템 성능을 향상시킬 수 있습니다.

5. CAN 브로드캐스트 동작을 재현하기 위한 멀티캐스트

CAN의 가장 대표적인 특징은 모든 메시지가 네트워크상의 모든 노드에 노출되는 '브로드캐스트(Broadcast)' 통신 모델이라는 점입니다. UDP 멀티캐스트(UDP Multicast)는 이러한 CAN 특유의 동작 방식을 이더넷 상에서 그대로 재현할 수 있게 해줍니다.

송신자가 특정 멀티캐스트 그룹으로 단 하나의 패킷만 전송하면, 해당 그룹에 가입(구독)된 모든 노드가 이 패킷을 동시에 수신하게 됩니다. 송신자가 일일이 여러 개의 점대점(Point-to-point) 연결을 만들 필요 없이, 네트워크 인프라(스위치)가 자체적으로 패킷 분배를 처리하는 방식입니다. 이를 통해 송신 노드의 부하를 줄일 수 있으며, 실제 물리적인 CAN 버스와 마찬가지로 효율적인 일대다(One-to-many) 통신을 구현할 수 있습니다.



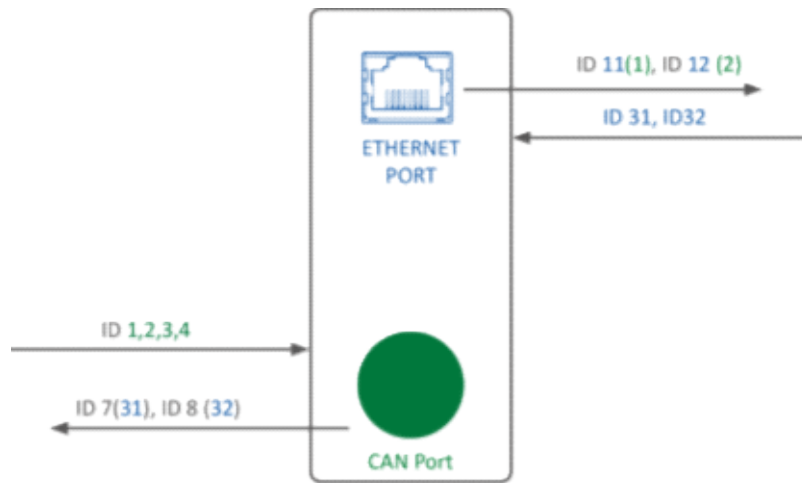
<그림 1: 모든 메시지가 모든 노드에 노출되는 UDP 멀티캐스트 기반의 브로드캐스트 통신>

6. 확장성 및 디커플링(분리) 매커니즘

이 아키텍처의 핵심 요소는 로컬(지역) ID 공간과 글로벌(전체) ID 공간을 서로 분리하는 것입니다. 로컬에서 중복되어 사용되는 CAN 식별자(ID)들은 매핑 매커니즘을 통해 글로벌 관점에서 고유한 ID로 변환될 수 있습니다. 유입된 CAN 프레임의 정보를 확장하거나 변환함으로써 전체 시스템 내에서 고유하게 식별되도록 보장하는 방식입니다. 실제 구현 시에는 전송되는 프레임 내에 고유한 장치 식별자(Device ID)를 포함시키거나, 표준 11 비트 식별자를 이더넷 백본용 고유 29 비트 확장 식별자로 변환하는 방식을 사용할 수 있습니다. 이를 통해 모든 노드가 내부적으로 0x100 과 같은 동일한 CAN ID 를 사용하더라도 충돌 없는 통신이 보장됩니다.

이와 동시에 데이터 분배는 선택적으로 제어됩니다. 존재하는 모든 메시지를 모든 노드로 무조건 전달하는 대신, 실제 관련이 있는 프레임만 선별하여 전송합니다. 이는 구성 가능한 필터링 매커니즘을 통해 이루어지며, 로컬 CAN 세그먼트에 과부하가 걸리는 것을 방지합니다. 실제 시스템에서는 게이트웨이 내부에 구독(Subscription) 목록이나 '관심(Interest)' 테이블을 두는 방식으로 이를 구현합니다. 각 노드는 자신에게 실제로 필요한 메시지 서브셋(부분 집합)만 수신하게 되므로, 로컬 CAN 인터페이스의 불필요한 부하를 줄일 수 있습니다. 또 다른 중요한 측면은 패킷 결합(Aggregation)을 통한 부하 감소입니다.

여러 개의 CAN 프레임을 하나의 UDP 패킷으로 묶어 보냄으로써 데이터 처리 오버헤드를 최소화하고 네트워크 자원을 더욱 효율적으로 활용할 수 있습니다.



<그림 2: 전체 시스템에서 고유하게 식별될 수 있도록 유입된 CAN 프레임을 확장하거나 변환하는 구조>

7. CAN 고유 특성의 유지

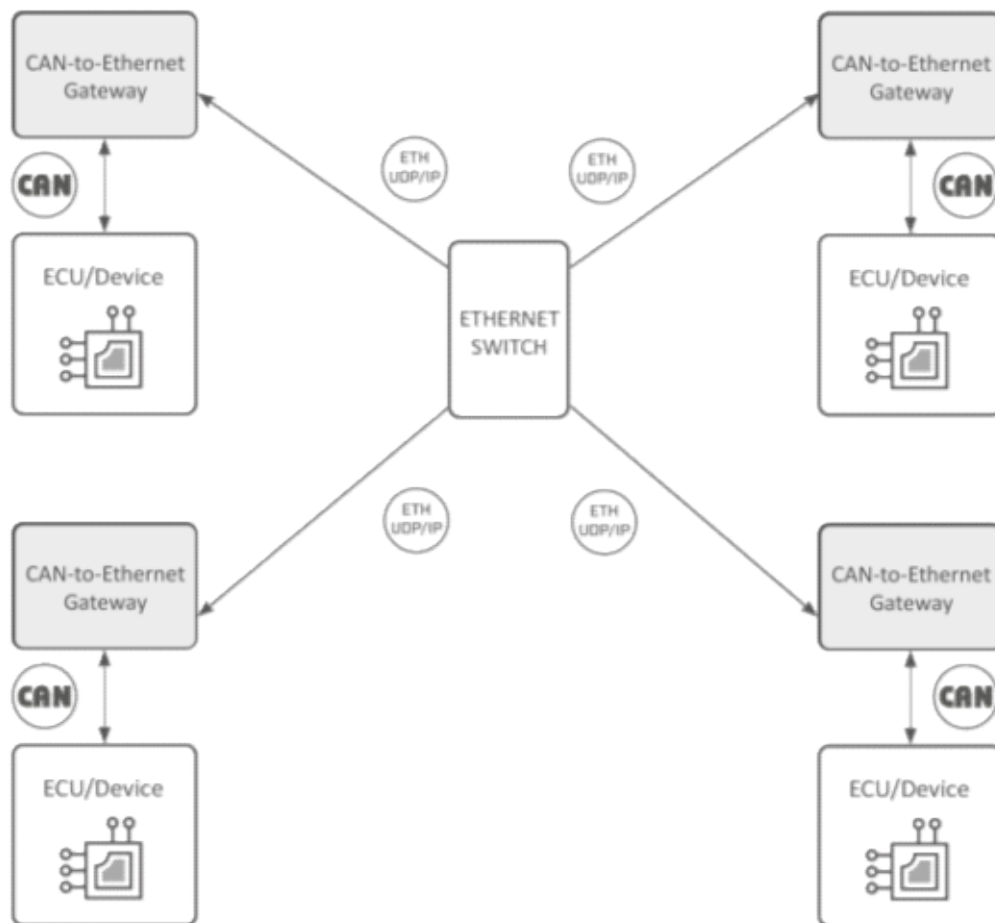
CAN 데이터를 이더넷으로 전송할 때는 프로토콜 고유의 핵심 특성들을 그대로 유지하는 것이 필수적입니다.

그중 결정적인 요소 중 하나는 바로 메시지 우선순위(Message prioritization)입니다. 기존 CAN에서는 버스 중재(Arbitration) 단계를 거치며 식별자(ID) 값에 따라 우선순위가 결정됩니다. 이더넷 환경에서는 이를 QoS(서비스 품질) 매커니즘이나 IP 우선순위 필드(DSCP 등)에 매핑하여, 우선순위가 높은 메시지가 네트워크 내에서 그에 걸맞게 우선적으로 처리되도록 보장할 수 있습니다.

타이밍 동작(Timing behavior) 역시 똑같이 중요합니다. UDP 패킷 내에 고해상도 타임스탬프(Timestamp)를 함께 심어서 전송하면, 수신 측에서 CAN 프레임이 원래 가지고 있던 시간적 선후 관계를 그대로 재구성해 낼 수 있습니다. 이는 지터(Jitter, 신호 전달 시간의 불규칙한 변동)를 최소화하고 시스템의 결정론적(Deterministic)인 동작을 유지하는 데 기여합니다.

마지막으로, 데이터 무결성(Data integrity, 데이터가 생성, 처리, 저장, 검색되는 전 과정에서

정확성, 일관성, 완전성, 유효성을 유지하고 손상, 변조를 방지하는 특성)은 계층화된 보호 가드레일을 통해 보장됩니다. CAN 자체의 CRC(순환 중복 검사) 매커니즘에 더해, 이더넷 전송 과정에서 UDP 체크섬(Checksum, 데이터 전송, 저장 과정에서 오류가 발생했는지 확인하기 위해 원본 데이터에 추가로 붙이는 검사값)이 추가적인 에러 검출 계층을 제공함으로써 데이터의 신뢰성을 한층 더 높여줍니다.



< 그림 3 : CAN-to-Ethernet 게이트웨이와 중앙 이더넷 스위치를 이용한 분산된 CAN 노드들의 연결 구조 >

8. 결론

UDP와 멀티캐스트의 조합은 기존의 Classical CAN 네트워크를 이더넷 환경으로 효율적으로 확장할 수 있는 길을 열어줍니다. 이 방식을 통해 물리적인 거리나 노드 수의 한계가 무너지고, 확장성은 크게 향상되며, CAN 통신만의 핵심적인 특성들은 그대로 보존됩니다.

결과적으로 이러한 접근 방식은 오늘날 산업 및 자동차 환경에서 요구되는 현대적인 분산형 시스템 아키텍처를 구축하기 위한 견고한 기반을 제공합니다.

이를 실제 시스템에 도입할 때, 앞서 언급한 매커니즘들을 내장하여 CAN과 이더넷 네트워크 간의 원활한 연결을 지원하는 전용 게이트웨이 솔루션들을 활용할 수 있습니다.

만약 이 기능을 직접 비즈니스나 프로젝트에 적용해 보고 싶으시다면, 아래 제품들을 참고해 보시기 바랍니다.

PEAK PCAN-Ethernet Gateway DR(IPEH-004010)

[\(주\)임베디드시스템코리아 쇼핑몰에 오신 것을 환영합니다!](#)

IXXAT CAN@net Basic

[Ixxat CAN@net Basic](#)