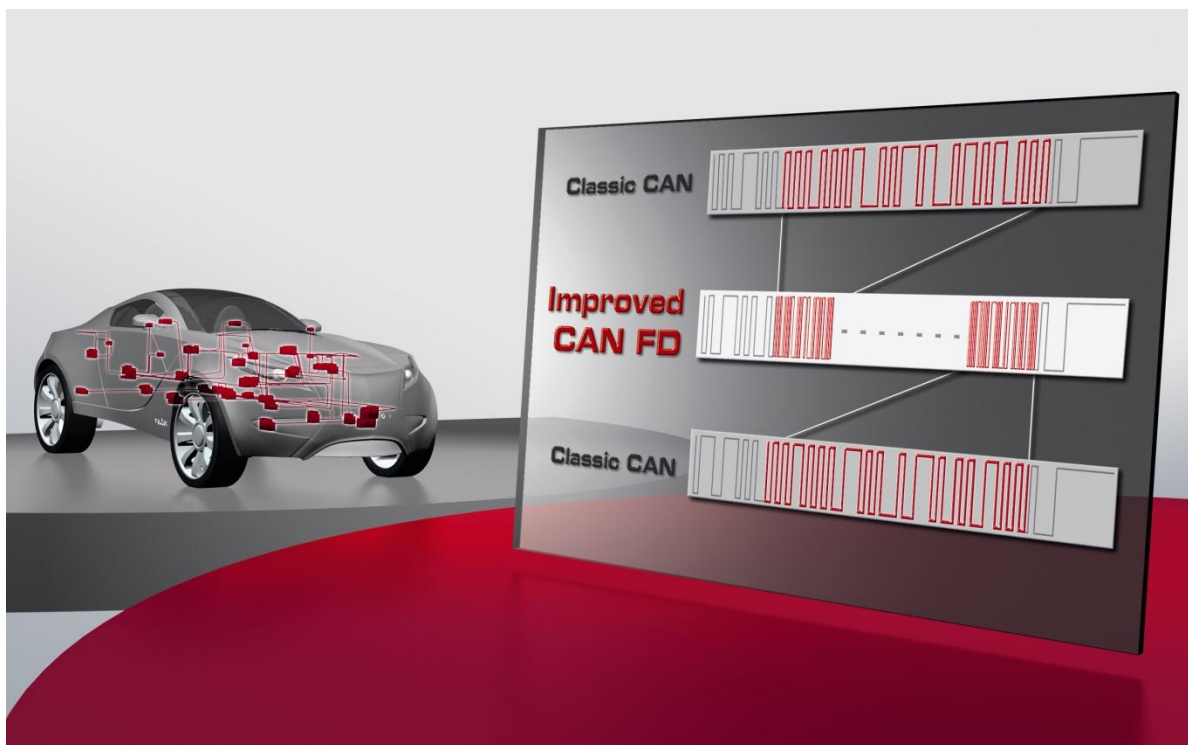


CAN 통신의 진화

기존 CAN 에서 개선된 CAN FD 로의 변화



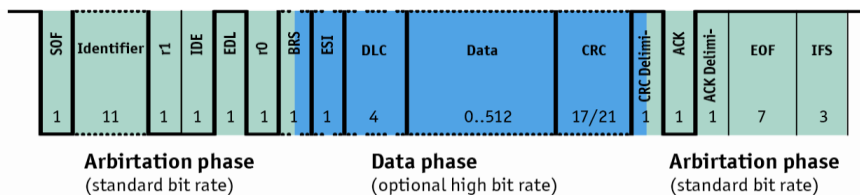
2012년 3월 Robert Bosch GmbH 는 새로운 CAN 프로토콜인 CAN FD(CAN with Flexible Data rate)를 공개하였다. 이 프로토콜의 주요 특징은 데이터 길이가 기존 8 바이트에서 64 바이트로 확장되고 데이터 전송 속도가 훨씬 더 빨라졌다는 것이다. 자료에 의하면 CAN FD의 성능은 High Speed CAN(1Mbit/s)과 FlexRay(10Mbit/s)의 사이에 있으며, 이러한 버스 시스템 간의 성능 차이를 저비용으로 메우기에 적합하다. 본 기사는 툴 공급자의 관점에서 무엇을 개선시켜야 하는지, 또 이러한 변화가 CAN FD 개발 및 시뮬레이션에 미치는 영향에 대해 다룬다. 하드웨어는 물론 데이터 포맷 및 각종 통신 계층에 이르는 다양한 영역에서 이러한 변화가 이루어지고 있다.

자동차 산업에서 CAN FD의 등장은 차량 내 전자장치의 데이터 트래픽 증가에 따른 병목현상을 해결하기 위한 보다 나은 네트워킹 솔루션의 새로운 기반이 되었다. 그동안 데이터 병목현상 때문에 비용 증가를 감수하더라도 한 네트워크를 다수 버스로 분할하는 등의 방법을 임시방편으로 사용해왔다. 그렇다고 CAN을 고성능의 FlexRay로 전환하기에는 너무 높은 투자비용이 필요하다.

또한, Event-driven 방식인 CAN 에 이미 익숙한 대부분의 개발자에게 Time-triggered 방식인 FlexRay 로의 전환은 작업 방식 및 사고에 큰 변화를 필요로 한다.

혁명 대신 진화

약 20 년 동안 자동차 산업에선 CAN 이 지배적인 버스 시스템이었기 때문에 많은 개발자들이 CAN 에 대한 충분한 노하우를 가지고 있다. 이러한 노하우는 CAN FD 프로젝트에도 적용될 수 있는데 CAN FD 는 버스 중재(bus arbitration), 메시지 응답(message acknowledgment), Event-driven 제어 등의 기존 CAN 컨셉을 그대로 사용하기 때문이다. 또한, CAN FD 의 장점인 빠른 비트 속도, 긴 데이터 길이를 활용하면 매우 유용하다. 예를 들면 트럭에서 긴 버스 라인 때문에 더 빠른 전송속도가 필요할 경우 64 바이트의 데이터 길이는 훨씬 더 많은 데이터를 처리할 수 있게 해준다.



EDL = Extended Data Length:
 CAN (dominant (0) = CAN frame
 Recessive (1) = CAN FD frame

BRS = Bit Rate Switch:
 CAN FD data phase starts immediately at sampling point of BRS:
 Dominant (0) = No change of bit rate for data phase
 Recessive (1) = Change to higher bit rate for data phase

ESI = Error State Indicator:
 Dominant (0) = CAN FD node is error active
 Recessive (1) = CAN FD node is error passive

그림 1: CAN-FD 프레임

비록 CAN FD 가 CAN 과 동일한 기능을 다수 공유하지만, 여전히 하드웨어 및 소프트웨어에 대한 개선된 프로토콜 확장 및 적응이 필요하다. 무엇보다도, CAN FD 에는 EDL(Extended Data Length), BRS(Bit Rate Switch) 및 ESI(Error State Indicator) 비트와 같은 3 개의 새로운 비트를 제어 필드에도 도입하였다. EDL 비트는 CAN FD 포맷의 프레임을 기존 CAN 포맷의 프레임과 구분하며, EDL 비트가 recessive(1)이면 CAN FD 프레임이고, dominant(0)이면 CAN 프레임으로 정의된다. 이와 유사하게, BRS 비트가 recessive(1)이면 데이터 필드를 전송할 때 더 빠른 비트 속도로 전환한다. ESI 비트는

CAN-FD 노드의 에러를 식별하는데 사용된다. 또한, DLC(Data Length Code)는 4 비트로 구성되어, 데이터의 길이를 12, 16, 20, 24, 32, 48 및 64 바이트 단위로 정의한다(그림 1).

툴 공급자의 직면 과제

소프트웨어 개발 툴 공급자는 고객이 CAN FD 와 같은 새로운 시스템을 도입하여 첫 ECU 개발을 시작할 때 이미 적절한 툴을 제공할 수 있어야 한다. CAN FD 를 기존 테스트 및 시뮬레이션 환경에 통합하기 위해서는 각종 하드웨어 레벨부터 애플리케이션 레벨에까지 신속한 변화를 요구한다. 또한, 통신 네트워크를 기술하기 위한 관련 데이터베이스가 필요하다. 이러한 맥락에서 CAN FD 모델링을 위한 최선의 방법은 무엇인지 또 64 바이트의 데이터 길이를 고려했을 때 PDU 개념이 필요한지 먼저 확인할 필요가 있다. PDU(Protocol Data Unit)는 흔히 FlexRay 와 함께 사용될 뿐만 아니라, AUTOSAR System Description 의 지원도 받는다. 아울러 추가적인 CRC 검사나 Tx 트리거로도 사용된다.

CAN FD 를 모델링 하는 데 있어서 CAN 이 가진 장점들을 가능한 많이 유지하는 것은 일리가 있다. 예를 들어, CAN FD 를 새로운 버스 시스템이 아닌 CAN 의 확장으로 취급한다면 기존 CAN 테스트 및 시뮬레이션 시스템을 신속하게 마이그레이션할 수 있다. 이에 따라 툴의 구성 변화도 상대적으로 작다. 테스트 스크립트 및 데이터베이스를 계속 사용하는 것도 가능하다. 만약 OEM 사 및 공급자가 초기에 CAN FD 를 8 데이터 바이트로 제한할 수 있다면 첫 프로젝트를 신속히 수행할 수 있을 것이다. DBC 데이터베이스는 CAN 응용 분야에서 널리 사용되고 있고 유사 산업 표준을 대표하는데 이 DBC 데이터베이스를 사용하는 것도 가능하다. 또한, DBC 는 J1939 와 같은 프로토콜에 사용되는 64 바이트 데이터를 처리할 수 있다.

FPGA 와 변경 가능한 트랜시버에 의한 플렉시블 하드웨어

소프트웨어 툴의 구조와 그 툴의 기본적인 분석, 테스트, 로깅 및 시뮬레이션 기능은 CAN FD 에 대한 적응이 거의 모든 계층의 자동차 프로토콜 스택에서 요구됨을 나타낸다. 물리적인 레벨에서,

툴은 버스 접근용 네트워크 인터페이스를 거의 항상 사용하는데 이때 네트워크 인터페이스는 FPGA 또는 표준 컨트롤러 칩에 기초하여 다양한 방식으로 실행된다. 컨트롤러 칩 기반은 경제적이지만 특정 기능 세트에만 사용할 수 있고 반드시 CAN FD 용으로 먼저 사용할 수 있어야 한다는 한계가 있다. 따라서 초기 CAN FD 단계에서 더 유연하고 신속히 사용 가능한 대안은 FPGA 기반의 인터페이스를 사용하는 것이다. 이 기술은 요구되는 리얼타임 요건을 충족할 뿐만 아니라, 드라이버 업데이트를 통해 64-바이트 지원 또는 버그 해결과 같은 새로운 기능을 제공한다. CAN FD 가 새로운 트랜시버를 필요로 하는지는 사용 용도에 따라 다르다. 예를 들면 차량 네트워크에서는, 기대속도가 2 와 4Mbit/s 사이이다. 그러나 플래싱 ECU 의 경우, 그 속도는 가능한 한 높아야 하며, 여기서 사용되는 CAN 트랜시버에 따라 8Mbit/s 이상도 나올 수 있다. Piggyback 트랜시버를 장착한 설정 가능한 하드웨어, 즉 소형 교환 가능 플러그-온 보드(그림 2)는 앞으로 트랜시버를 지원할 유연한 접근법을 제공한다.



그림 2: Piggyback 트랜시버를 가진 CAN-FD 버스 인터페이스

통신 레벨에서, 개발자는 흔히 프레임을 묘사하고 그 프레임의 특정 데이터로 접근할 수 있는 분석 툴이 필요하다. 여기서 새로운 CAN FD 비트 EDL, BRS 및 ESI 를 정확히 해석하는 것이 절대적으로 필요하고, 이는 확장된 유용한 데이터 (그림 3)에도 동일하게 적용된다. 프레임을 툴에 의해 발송된 이벤트로 모델링함으로써 간편하게 필요한 사항들을 변경할 수 있다. 즉, 기존 CAN 이벤트에 CAN

FD 비트를 더해 유용한 데이터의 길이를 더 길게 확장할 수 있다. 오직 메시지를 표시하는 분석 창만 CAN 프레임과 CAN FD 프레임으로 구분할 필요가 있다.

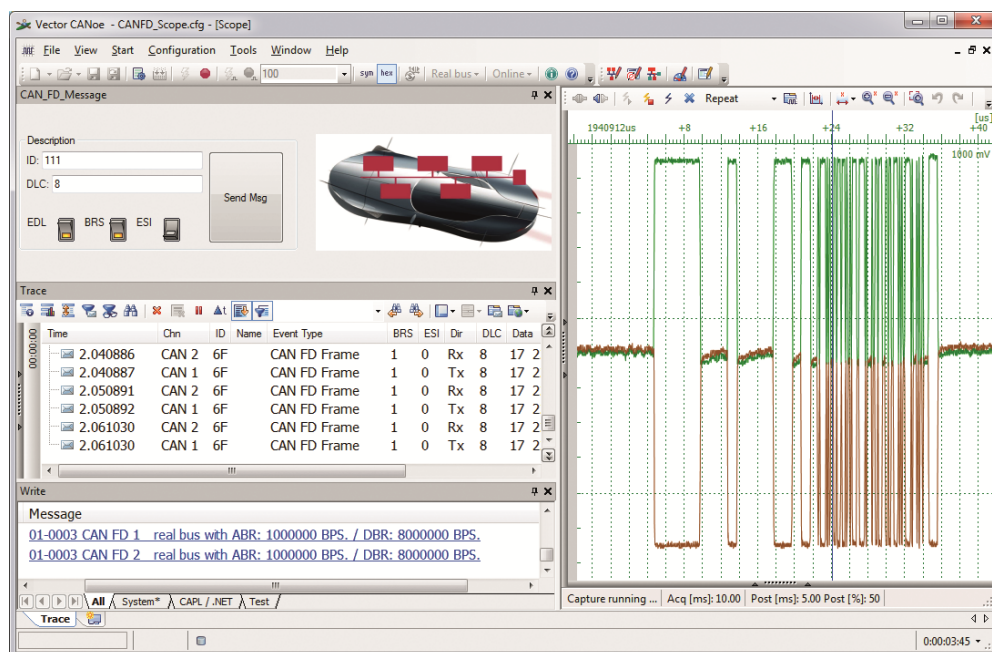


그림 3: CAN-FD 시뮬레이션 및 테스트 툴

상위 계층에서의 CAN FD

상위 계층에서는 진단, 네트워크 관리(NM), 전송 프로토콜(TP), 상호작용 계층(IL) 및 Tx 모델로 초점이 맞춰진다. 전송 계층에 대한 적용은 훨씬 더 많은 프레임을 해석하는 반면, 적용 계층은 일반적으로 프레임 대신 신호로 작업한다. 네트워크 관리, 상호작용 계층 및 Tx 모델은 모두 특정 자동차 OEM 요건에 따라 달라진다. CAN FD의 확장된 데이터는 상호작용 계층 및 전송 계층 모두에 영향을 미치므로 변경이 필요하다. 적절한 인터페이스를 가진 모듈식 툴의 이점은 OEM 고유 특징이 실행 또는 교환하기 쉽다는 것뿐만 아니라 CAN FD에 요구되는 확장도 실행하기 쉽다는 것이다. 적용 수준에서 개발자들이 시작단계에서 적용을 특정 프로토콜에는 독립적으로 신호에는 의존하도록 설계하였다면 시뮬레이션 모델, 시험 스크립트 및 그래픽 사용자 인터페이스를 거의 변경하지 않고 계속 사용할 수 있다.

CAN (FD), LIN, FlexRay 의 새로운 판도

새로운 버스 시스템의 도입은 기존 네트워크 프로토콜에 판도 변화를 가져온다. CAN FD 는 기존의 LIN 에 더 큰 편익을 제공하지는 않기 때문에 LIN 의 편의성은 그대로 유지된다. 그러나 단순히 CAN 프레임의 데이터를 LIN 프레임에 복사하여 8 바이트보다 더 큰 데이터로 단순하게 TP 라우팅(또는 "raw TP")하는 것은 불가능하다. FlexRay 의 경우 LIN 과 상황이 다르다. CAN FD 는 이벤트 구동식 응용 시, FlexRay 보다 경제적인 대안으로 선호되기 때문이다. 반면에 FlexRay 는 Real-time application) 또는 Deterministic application 으로 사용이 제한된다. 일부 마이그레이션은 기존 CAN 시스템 내에서도 이루어질 것이다. 약 50 퍼센트 높은 버스 부하를 가진 네트워크가 그러하다. CAN FD 는 높은 대역폭을 통해 높은 버스 부하를 가진 네트워크를 분할하지 않으며, 역으로 다중 네트워크를 단일 네트워크로 합쳐서 사용할 수 있다. 이를 통해 불필요한 게이트웨이의 감소로 인한 비용 절감과 원치 않는 게이트웨이의 반응 시간(latency)을 제거할 수 있다.

현재로서는 CAN 및 CAN FD 가 혼합된 네트워크를 사용할지에 대해 아직 일치된 의견은 없고, 이는 차량 제조사의 결정에 달려있다. CAN 및 CAN FD 를 혼합해서 사용하려면 어떤 조치를 해야 한다는 점은 분명하다. 예를 들어, FD 운용에서 CAN 노드는 수동으로 분리되어야 한다는 것이다. 그렇지 않을 경우 EDL 비트 때문에 형상 에러를 탐지해야 하고 에러 프레임이 전송된다.

CAN 에서 CAN FD 로의 마이그레이션

진보된 분석, 시험 및 시뮬레이션 툴은 CAN FD ECU 및 네트워크 개발과 관련된 거의 모든 상황에서 유용하다. 개별 네트워크 노드를 분석하는 것에서부터 전체 네트워크를 시뮬레이션하는 등 사용 영역도 다양하다. 이러한 툴은 개발자가 버스 부하를 예측하고 시뮬레이션 데이터를 기반으로 최적의 결정(예: 네트워크 분할 필요 여부 결정)을 내릴 수 있도록 돕는다. 툴은 개발 과정에 걸쳐 통신 버스 시뮬레이션을 수행하는데 사용되며, 이는 미 구현된 서브네트워크와 컴포넌트를 대신한다. 시뮬레이션 된 노드는 단계적으로 실제 ECU 와 교체된다. 또한, 시뮬레이션은 게이트웨이 역할을 할

수도 있으며, 기존 CAN 네트워크와 새로운 CAN FD 네트워크에 연결될 수도 있다. 이때 기존 CAN ECU에는 어떠한 변화도 줄 필요가 없다.

요약 및 결론

가변형 FPGA 기반 네트워크 인터페이스와 고성능 분석, 테스트 및 시뮬레이션 툴을 바탕으로 자동차 OEM 및 공급업체는 성공적인 CAN FD 개발에 필요한 모든 요소를 갖추고 있다. 앞에서 설명한 시스템은 여러 데이터베이스 포맷을 지원하며, 신호 및 PDU 레벨에서 다양한 추출은 물론 오픈 인터페이스를 통해 OEM 별 특성을 구현할 수 있다. 네트워크 시뮬레이션 및 통신 버스 시뮬레이션은 개발과정 중 발생하는 모든 상황에서 최적의 테스트 조건을 보장한다.

독일 출판물 Hanser automotive” 2013 년 2 월 발행 기사 번역판

그림 제공:

Vector informatik GmbH

참고 문헌:

CAN-FD specification V1.0, Robert Bosch GmbH

링크:

Vector CAN Solutions: www.vector.com/can_fd

저자:



피터 데커 Peter Decker (Dipl.-Ing. (FH))

2002 년 이래 벡터 인포마틱(Vector Informatik)에 근무해 왔으며, 현재 CANoe/CANalyzer 용 다중 버스개발 분야의 제품 매니저이다.

본 자료 배포 시 최종 인쇄물을 당사에 보내주시면 감사하겠습니다.
배포와 관련하여 문의사항이 있으시면 언제든지 연락주시기 바랍니다:

벡터코리아 편집자 연락처:

마케팅팀 김용성 매니저

서울특별시 용산구 한남대로 11 길 12 고와스빌딩 5 층

Tel. 02-807-0600 Ext.5009, Fax. 02-807-0601

E-mail: yongseong.kim@vector.com