

기존의 차량 시험 작업실에 FlexRay 제어장치 적용 사례

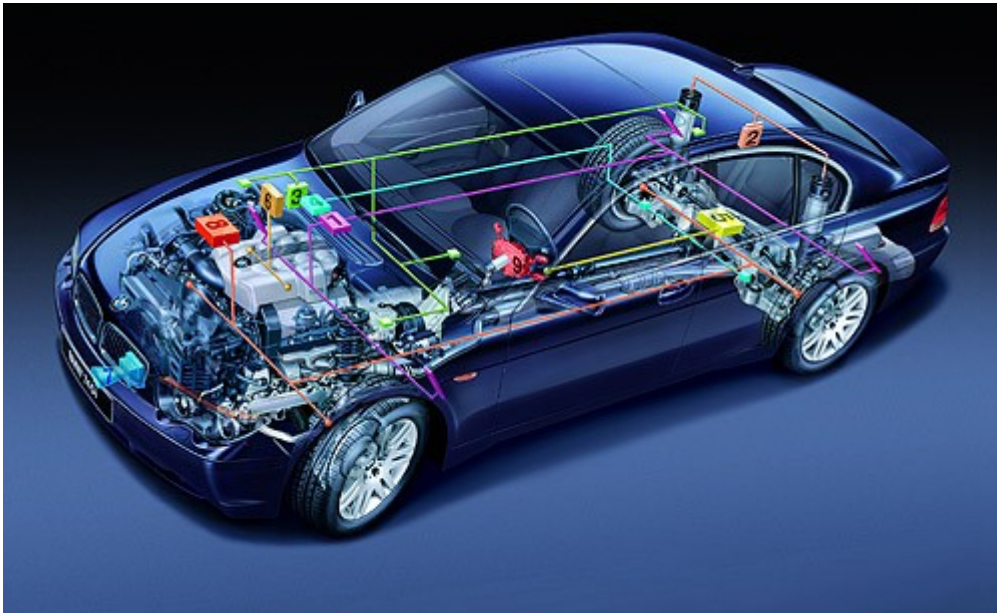
글쓴이 :

Roland Bacher, BMW Group

Herbert Haas, BMW Group

Martin Simons, IXXAT Automation

오늘날, 차량 시험 작업실에서의 측정과 제어 기술은 CAN 인터페이스를 통해 제어 장치들에 연결되어 있습니다. FlexRay 적용이 늘어남에 따라, BMW 그룹은 FlexRay를 기존의 차량 시험 작업실에 통합 시키기 위한 포괄적인 구상을 개발해왔으며 성공적으로 구현하게 되었습니다. 이 새로운 구상으로 FlexRay 사용자와 개발자들이 기존의 플랫폼을 교체하지 않고도 FlexRay 제어장치를 적용할 수 있게 되었습니다.



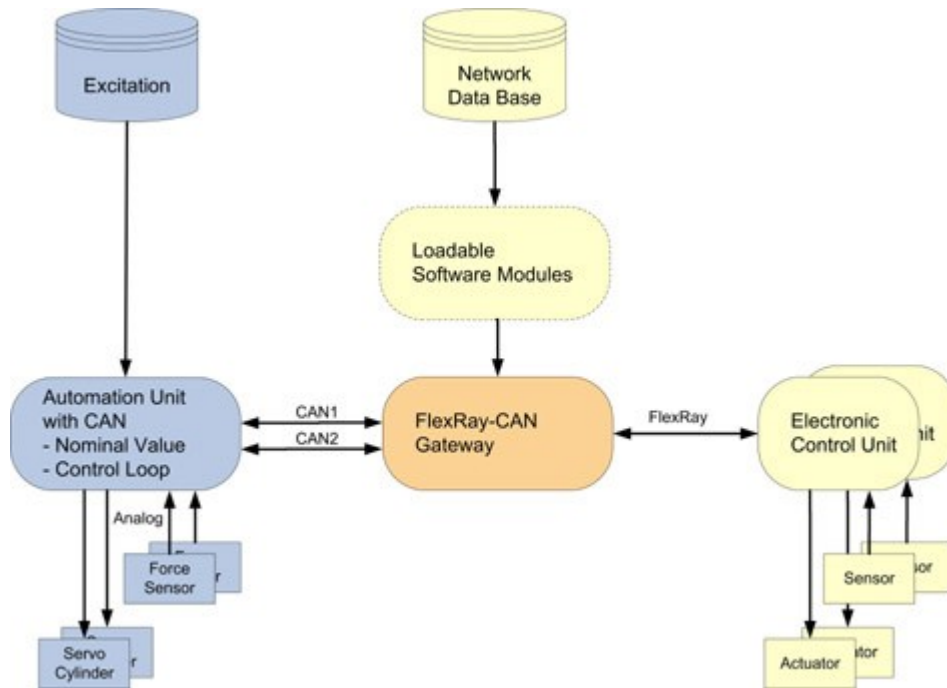
BMW 그룹은 FlexRay를 BMW X5에서, 매우 강력한, 실시간 통신 시스템으로 도입한 최초의 자동차 제조업체입니다. 또한 앞으로 계획되어 있는 모델들은 FlexRay를 포함하게 될 것입니다. FlexRay는 CAN 보다 20배가 큰 대역폭을 지닌 결정적 통신 시스템(deterministic communication system)이며 차량에서의 제어 유닛, 센서 그리고 액추에이터들 간의 늘어나고 있는 데이터 전송 요건들을 충족시킬 수 있습니다.

FlexRay의 도입은 BMW 그룹의 여러 부서들의 시험 작업실에 상당한 효과를 가져오고 있습니다. 오늘날 시험 작업 자동화는 CAN 기반이며, 차량에서의 전기 시스템과 시험 작업에서의 자동화 유닛은 CAN 인터페이스를 통해 통신합니다. 그러나, 새로운 BMW X5에서는, 한 예로, 다양한 충격 흡수 조정 (shock absorber adjustment) 제어 유닛들이 이미 FlexRay 버스를 통해 통신하고 있습니다. 기존의 시험 작업에 포함된 차량 요소들과 마찬가지로 이러한 FlexRay 제어 유닛들을 함께 시험하기 위해서는, FlexRay 네트워크에서 전송된 메시지들은 CAN 인터페이스로 변환되어야 했습니다. 그래야 기존의 시험 작업들을 그들의 특수한 측정 그

리고 제어 기술과 함께 계속해서 사용하는 것이 가능했던 것입니다.

이를 가능토록하기 위하여, FlexRay 메시지를 CAN 메시지로 변환하고 그 반대도 가능한 자유롭게 구성 가능한 게이트웨이 개념이 BMW 그룹에서 개발되었습니다.

아래그림은 시험 작업에서 FlexRay-CAN 게이트웨이의 통합 예를 보여줍니다.



자동화 유닛은 서보 실린더(servo cylinder)에 대한 타깃 값을 정의하고 센서를 통해 제어합니다. 서보 실린더는 포함된 차량 요소들의 로드(load)를 시뮬레이트합니다. 제어 유닛들은 그들의 관련 액추에이터와 함께 그들의 센서에 의해 등록된 로드(로드)에 반응하며 FlexRay 버스에서 메시지를 보내는데, 자동화 유닛은 어느 때와 같이 CAN 버스를 통해 수신합니다. 이것은 FlexRay-CAN 게이트웨이를 통해 이루어지며, 이 게이트웨이가 FlexRay 버스에서 제어 유닛의 메시지를 수신하고 이것들을 CAN 버스를 통해 자동화 유닛으로 전달합니다.

FlexRay-CAN 게이트웨이의 주요 필수요건은 다음과 같습니다:

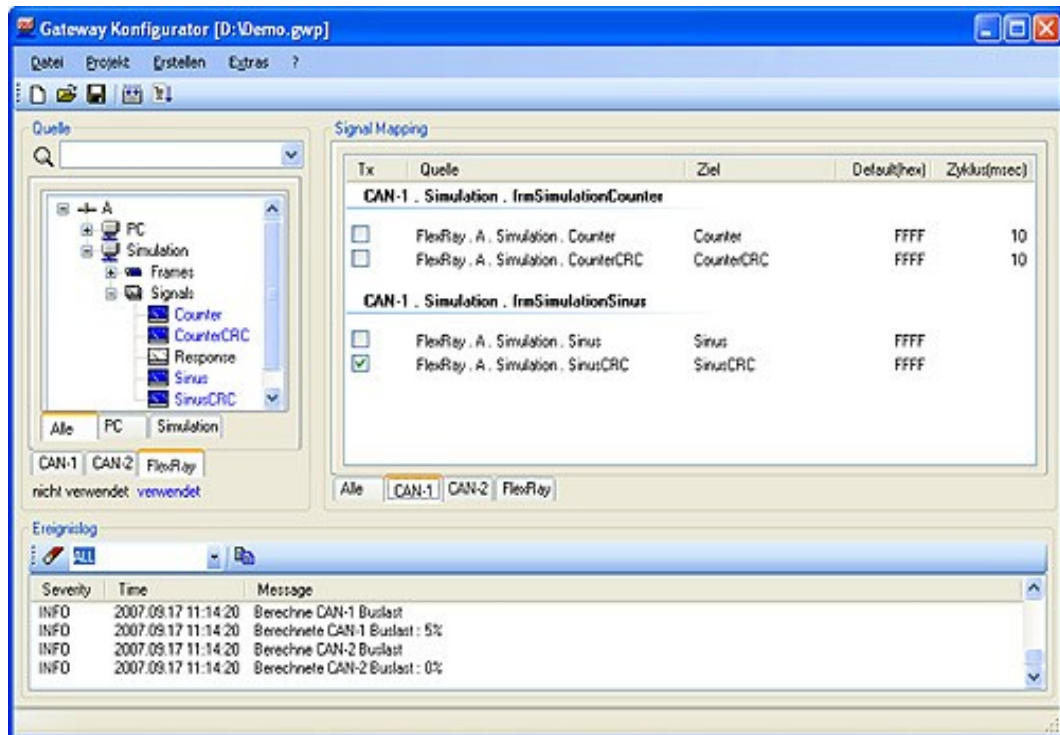
- CAN 인터페이스가 장착된 자동화 기술을 통합할 수 있는 기능
- 다양한 제어 유닛들에서의 사용을 위한 구성 가능 능력
- 자립형(stand-alone) 작동, 즉 구성 후에 PC 없이 작동
- 가장 최근에 저장된 구성을 이용한 자동 시동
- 필요한 데이터 보안 정보를 포함한 메시지 생성
- 자동화 유닛과 호환 가능한 Command/status 인터페이스
- FlexRay의 버스 신호를 CAN 버스로 그리고 그 반대로도 변환하도록 정의된 시간 응답
- 이더넷 인터페이스를 통해 PC에서 장치를 구성

구성 툴의 사용자 인터페이스는 FlexRay 버스에서의 신호를 CAN버스의 신호로 간단히 지정할 수 있어야 합니다 (그 반대의 경우도 마찬가지). 이를 위해, 이 툴은 CAN 버스 (FIBEX 또는 CANdb를 사용)와 FlexRay 버스 (대개 FIBEX를 사용)에서의 메시지 트래픽의 description 파일들을 읽을 수 있어야 합니다. Description 파일이 없는 CAN 버스들을 위해, 이것은 선택 사항입니다. 이 툴은 일관성(consistency)을 위해 사용자의 모든 인풋들을 확인해야 합니다 (예. 비트 길이, 데이터 유형). 구성의 모든 파일들은 하나의 프로젝트에 포함되어야 합니다.



IXXAT의 FlexRay CCM 은 하드웨어와 소프트웨어 플랫폼으로 사용됩니다. 이 유닛은 이미 게이트웨이로써의 필수 요건의 대부분을 포함하고 있습니다. 견고한 외관, 폭넓은 온도 범위, 다양한 범위의 공급 전압 덕분에, 이것은 특히 시험 작업실과 차량에서의 사용에 적합합니다. 다양한 업체들의 분석과 진단 툴들 그리고 원래BMW 그룹에 의해 개발되었던 잔여 버스 시뮬레이션이 이미 FlexRay CCM에서 실행되고 있습니다. 이러한 모든 애플리케이션과 함께, 읽어올 수 있는 소프트웨어 모듈을 통해 FlexRay CCM의 하드웨어에서 기능들이 실행됩니다. 또한 IXXAT는 상대적으로 저렴한 개발 비용으로, 읽어오기가 가능한 또 다른 소프트웨어 모듈과 이에 따른 PC-기반 구성 툴을 이용한 FlexRay-CAN 게이트웨이를 실행할 수 있었습니다.

일반적으로, 사용자는 메시지 그 자체가 아니라 FlexRay와 CAN 메시지에 전달된 처리 변수들로 작업합니다. 이러한 이유로, 신호 할당으로부터의 메시지 변환은 PC에서 계산되며 읽어오기가 가능한 표로 제공됩니다. 게이트웨이 구성 툴은 아래 그림에서와 같이 간단히 "끌어 당기기"로 신호 할당이 가능합니다. 이것은 신호들을 관련 이름과 함께 제시함으로써 사용자의 인풋을 지원합니다. CAN 버스를 위한 description 파일이 없다면, 사용자가 직접 메시지의 변환을 기입할 수 있습니다.



Lodable gateway 소프트웨어 모듈은 FlexRay 버스에 있는 메시지들을 CAN 버스의 메시지들로 변환하며 그 반대의 경우도 가능합니다.

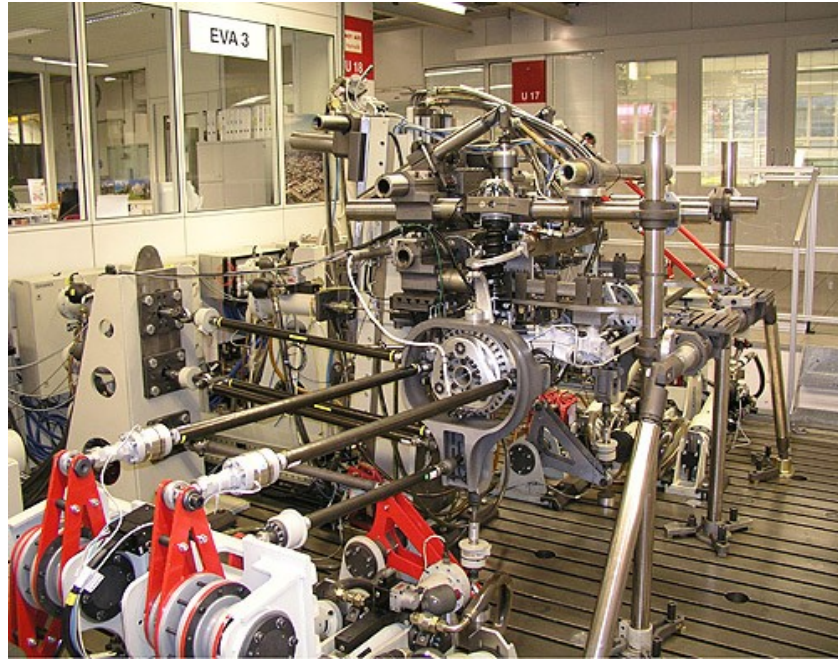
FlexRay 버스는 자동으로 생성된 코드에 의해 작동하는데, 이것은 제어 유닛이 예상하는 데이터 백업 정보와 함께 특정 사이클에서 FlexRay 메시지를 전송합니다. FlexRay-CAN 게이트웨이가 이에 따라 구성되면, 이것은 자동으로 FlexRay 버스를 시작합니다.

CAN 버스가 메시지와 같이 제공됩니다: 사건-제어식 또는 주기적으로 가능하며, 주기 시간(cycle time)은 구성하는 동안 입력됩니다. CCM의 양쪽 CAN 버스들이 사용될 수 있습니다. 일반적으로, FlexRay 버스에서 단지 몇 개의 신호들만 관련이 있기 때문에 CAN 버스에서의 대역폭은 충분합니다. 대역폭이 초과한다면, FlexRay-CAN 게이트웨이가 이것을 메시지로 보고합니다.

다운로드 대화상자를 이용하여, 사용자는 필요한 파일들과 코드를 FlexRay CCM의 하드웨어에 로드합니다. 그러면 FlexRay-CAN 게이트웨이는 자립형 (stand-alone) 모드에서 작동합니다. PC와의 연결은 더 이상 필요하지 않습니다. 스위치를 켜면, FlexRay-CAN 게이트웨이는 가장 최근에 저장된 구성으로 시작합니다. 사용자는 단지 유닛을 PC에 연결 하고 변경을 할 수 있도록 구성 툴을 시작하면 됩니다.

자동화 유닛의 command/status 인터페이스는 FlexRay-CAN 게이트웨이의 CAN 인터페이스 두 개 중의 하나를 통해 실행되며, 이것은 Reset 명령을 통해 수신하고 주기적 상태(cyclical status) 메시지를 전송합니다. 현재 BMW 그룹은 다양한 검사 기법들에서 FlexRay-CAN 게이트웨이를 다음과 같이 성공적으로 사용하고 있습니다:

- 내구성 강도 시험 작업 (아래 사진)
- 엔진과 기어 시험 작업
- Functional development drive
- 작업량을 결정하기 위한 차량 측정 기법
- 고객 행동 측정 기법



이 만능 게이트웨이는 BMW 그룹의 부품 제조업자들에 의해서도 사용되고 있습니다.

FlexRay-CAN 게이트웨이의 개발과 함께, 광범위한 기존 검사 장비의 지속적인 사용과 CAN 기반의 시험 작업실을 FlexRay기반 시스템으로의 빠른 변환을 가능하게 하는 솔루션이 상대적으로 저렴한 개발 비용으로 구현되었습니다.

