

IXXAT사의 툴을 이용한 CANopen의 구현

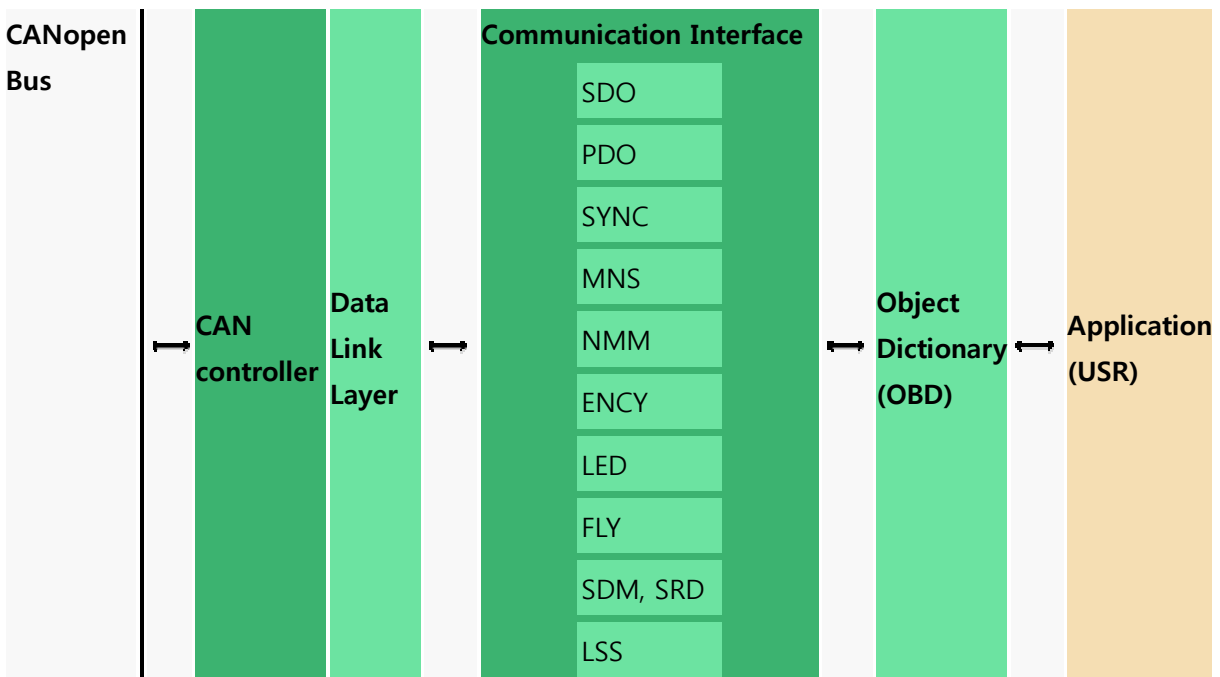
임베디드장치에서 CANopen 을 구현하기 위해서는, 프로토콜 스택이 사용되는데, 이는 일반적으로 기업 라이선스 형태로 제공됩니다. IXXAT 는 검증된 CANopen 프로토콜 스택을 C-code 또는 기존의 여러 마이크로컨트롤러에 적용된 형태로 제공합니다. CANopen 스택은 "Slave" 또는 "Master/Slave"로 이용할 수 있습니다. 용어 "Slave"와 "Master"는 네트워크 관리와 관련하여 서로 다른 기능을 말하는데, 이것은 논리적 Master/Slave 관계의 형태로 CANopen 에서 구현됩니다.

CANopen 스택은 통신, 네트워크 관리, 일반적인 시스템 서비스들, 그리고 장치 서술에 필요한 모든 기능을 개발자에게 제공합니다. 따라서 판매되는 프로토콜 스택을 구매하여 구현 한다는 것은 애플리케이션의 기능을 프로토콜 스택에 연결 하기만 하면 된다는 것을 뜻합니다.

CANopen 프로토콜 소프트웨어의 구조

IXXAT의 CANopen 스택은 Network Management (NMT), Process Data Objects (PDO), Service Data Objects (SDO), Emergency (EMCY), Synchronization (SYNC), SDO-Manager (SDM/SDR), Layer Setting Services (LSS) 와 Object Dictionary (OBD)같은, CANopen의 다양한 서비스들에 맞춰 확장 가능한, 모듈식으로 구성 되어있습니다. 모든 모듈들은 객체사전에 대한 액세스를 갖는데, 이는 통신 인터페이스와 애플리케이션의 데이터, 파라미터, 기능들을 결부시켜 소프트웨어 패키지의 핵심적인 인스턴스(instance)를 표현합니다.

CAN 컨트롤러에 대한 액세스 (예. 메시지 전송과 수신을 위한)는 "Data Link Layer" 인터페이스를 통해 실행되며, 그것은 또한 별도 모듈 (DLL-module)에 통합됩니다. 이것은 다른 CAN 컨트롤러 구현에 대한 CANopen 프로토콜 소프트웨어의 빠르고 간단한 적용을 가능하게 합니다.



간결한 구성과 최적의 메모리 요건

IXXAT CANopen 스택의 기능은 애플리케이션의 요건에 적합하게 쉽게 수정될 수 있습니다. 이를 위해, 개발자가 할 일은 중앙구성파일의 "defines"를 이용하여 functions/modules을 단지 활성화 또는 비활성화 하는 것뿐입니다.

16비트 마이크로컨트롤러에서 무난한 범위의 유용한 기능들을 가진 슬레이브를 구현하기 위해서는, 약13k ROM 과 1k RAM의 메모리 조건이 표준적인 값으로 여겨집니다. 만약 개발자가 이보다 적은 자원들만 이용 한다면, 이러한 값들은 구체적인 수정을 통해 더욱 축소될 수 있습니다.

중요한 것은 높은 수준의 유연성, 종합적인 테스트와 신뢰할만한 장기간의 유지보수, 그리고 추가적인 개발 능력입니다.

IXXAT 는 서로 다른 플랫폼에서 실행 가능한(cross-platform) C-소스코드 또는 이미 특정 마이크로컨트롤러와 CAN 컨트롤러에 적용된 CANopen 프로토콜 소프트웨어를 제공합니다. 따라서 개발자들은 마치 자신이 개발한 것처럼, 필요한 대로 마음껏 소프트웨어를 수정하고 관리할 수 있습니다.

CANopen 표준화 작업 시작 이후, IXXAT는 활발하게 CANopen 협회에서 활동하고 있기 때문에, 초기 단계부터 최종버전 규격의 확실한 구현이 가능합니다. 또한 IXXAT는 CANopen 스택의 꾸준한 개발과 유지보수에도 상당한 노력을 기울이고 있습니다. 소프트웨어 유지보수(maintenance) 계약을 통해, 고객들은 IXXAT의 지속적인 기술지원의 혜택을 누릴 수 있습니다.

고객들에게 고품질의 제품을 공급하기 위하여, IXXAT 는 CiA의 최신 검증 테스트 프로그램을 통하여 항상 자사의 프로토콜 스택을 검사하고 있습니다. 이것은, IXXAT 소프트웨어가 수년간 현장에서 성공적으로 사용되고 있다는 사실과 더불어, 우리의 고객들에게는 그들 제품에서의 성공적인 CANopen의 무결점 구현과, 장기적인 유지보수(maintenance)를 확실하게 책임집니다.

개발할 것인가? 아니면 구매할 것인가?

규격(Specification)들이 라이선스 비용 없이 무료로 이용 가능하며, 현재 여러 가지 공개된 source initiatives가 있기 때문에, 종종 CANopen 프로토콜 소프트웨어를 스스로 개발 하는 것을 검토하는 개발자들이 있습니다. 물론 이것은 흥미로운 도전입니다. 그러나 경제적인 관점에서는 그다지 해볼만한 것이 아닌데, 왜냐하면 요즘은 완성된 CANopen 프로토콜 소프트웨어를 매우 경제적인 가격으로 많은 업체들로부터 제공 받을 수 있기 때문입니다.

효율적으로 검증되고 문서화된 CANopen 프로토콜 스택을 개발하기 위해서는, 규격의 모든 기능이 구현되지 않는다 하더라도 적어도 4~6 개월의 개발 기간이 예상됩니다. 그러나 "초기구현"이 정말 효율적으로 최적화 되었는지, 무결점 제품인지, 그리고 직접적이지 않은

CANopen 규격의 “미묘함”들이 정확하게 구현 되었는가에 대한 과제가 남습니다. 무료 또는 공개된 구현들은 대부분 교육용 또는 비-상업용 환경을 염두에 두기 때문에 거의 이러한 현실을 해결하지 못합니다.

CANopen의 구현에 무엇이 포함되는가?

IXXAT 는 자사의 CANopen 프로토콜 스택과 함께 종합적인 문서와 예제프로그램을 제공합니다. Project 파일들은 예제 프로그램들과 같이 제공되어 컴파일러 제조업체의 관련 개발환경에 직접적인 통합이 가능합니다. 모든 예제 프로그램들은 상용 플랫폼(evaluation board 또는 IXXAT 인터페이스 카드)에서 직접 실행될 수 있습니다.

구현은 다음과 같이 실행됩니다:

- 타겟 플랫폼의 하드웨어에 맞춰 변경 (Timer, Interrupt system)
- Object dictionary entries 의 생성
- Configuration file 변경
- 컴파일과 검사. 검사를 위해 예제 응용 프로그램이 타겟 플랫폼으로 전송될 수 있습니다. 그러면 이것은 사용자 응용프로그램에 대한 기반으로 사용될 수 있습니다.

타겟 시스템에 대한 초기의 CANopen 구현은 수일 내에 이루어질 수 있습니다.

그렇지 않다면, IXXAT에서 구현을 실행할 수도 있습니다. 이 경우, 필요 조건들을 고객과 함께 지정하고 난 후에, IXXAT에서 구현합니다. 이 구현은 소프트웨어에 정통한 숙련된 엔지니어들에 의해 이루어지기 때문에, 사용자의 독자적인 구현에 비해 저비용의 빠른 대안이 될 수 있습니다.