

AUTOSAR 기반의 FlexRay 모듈 개발

주 흥 택[†] · 정 설 영^{††} · 이 명 숙^{†††}

요 약

AUTOSAR는 자동차용 공개 소프트웨어 플랫폼으로서 차세대 자동차용 네트워크 표준인 FlexRay 모듈을 포함하고 있다. 본 연구는 AUTOSAR 규격을 기반으로 FlexRay 기본 소프트웨어 모듈들인 FlexRay State Manager, FlexRay Transport, FlexRay Interface, FlexRay Transceiver Driver, FlexRay Driver 모듈들을 개발하였다. 또한 개발 과정에서 얻은 설계와 구현 방법을 함께 제시하고, 개발된 모듈들을 마이크로컨트롤러에 탑재하여 동작을 검증하였다.

키워드 : 오토사, 플렉스레이, 자동차 내부 네트워크

Development of FlexRay Module based on the AUTOSAR

Ju, Hong-Taek[†] · Jeong, Seol-Young^{††} · Lee, Myung-Suk^{†††}

ABSTRACT

The AUTOSAR(AUTomotive Open System ARchitecture) is an open software platform for automotive. The AUTOSAR contains the modules of FlexRay that is the next generation automotive network protocol. We have developed the modules including FlexRay State Manager, FlexRay Transport, FlexRay Interface, FlexRay Transceiver Driver and FlexRay Driver based on the standard of the module. This thesis presented not only the result of development but the result of design and implementation. And the developed modules were ported to the microcontroller board and verified.

Keywords : AUTOSAR, FlexRay, In-Vehicle Network

1. 서 론

최근 자동차의 전자제어 시스템은 센서와 제어를 관장하는 ECU(Electronic Control Unit)가 핵심 역할을 하고 있다[1]. 많은 ECU가 자동차에 사용되면서 ECU에 탑재되는 소프트웨어가 방대해지고, 담당하는 역할이 중요해졌으며 자동차 소프트웨어의 결함 감소, 개발 비용 및 개발 기간 단축 등을 위하여 소프트웨어 개발 방법과 관리에 대한 관심이 증대되고 있다[2].

자동차 소프트웨어 개발 방법을 통일하고 관리 방법을 단순화하기 위하여 주요 자동차 업체들은 AUTOSAR(AUTomotive Open System ARchitecture)[3]라는 협력체를 구성하여 자동차 소프트웨어 표준 규격을 제정하고 있다. 이 표준은 일관성 있는 설계 방법과 호환성 있는 인터페이스의 규격을 제시하고 있다. 또한 AUTOSAR에서는 FlexRay가 핵심 네트워크 기술로 제시되어 있고 이를 어떻게 활용할지에 대하

여도 자세히 기술되어 있다[4]. FlexRay[5]는 실시간성과 높은 대역폭, 높은 신뢰성을 가진 기술로서 최근 표준화가 완성되어 자동차에 적용되기 시작하였다. 이러한 FlexRay는 상업용 패키지로 개발되어 판매되고 있으나 현재까지 개발 결과가 학술논문으로 발표된 바가 없다. FlexRay는 차세대 자동차용 네트워크 기술로서 앞으로 많은 기술 발전을 필요로 하며 이러한 기술발전은 학술활동을 통하여 공개적으로 이루어져야 한다. 물론 업체에서도 FlexRay 제품을 개발하면서 FlexRay 기술을 발전시킬 수 있으나 이와 같은 발전은 공개되기도 어려워서 독점적인 기술이 될 수 있으므로 공개표준인 FlexRay의 발전에 크게 도움을 주지 못한다.

따라서 본 연구에서는 먼저 AUTOSAR에서 제시하고 있는 규격들을 분석하였다. 그리고 분석된 표준에 따라서 개발하기 위한 방법을 제시하였으며, 이 방법에 따라서 FlexRay 네트워크의 기본 소프트웨어 모듈인 FlexRay State Manager, FlexRay Transport, FlexRay Interface, FlexRay Transceiver, FlexRay Driver 5가지 모듈들을 개발하였다. 또한 개발된 기본 소프트웨어 모듈들은 보드에 직접 포팅하여 기본 통신 동작을 확인하여 검증 과정을 거쳤다. 본 논문에서는 개발결과만 제시하는 것이 아니라 FlexRay 모듈 개발과정에서 얻어진 설계와 구현 방법을 제시하여

※ 본 연구는 2003년도 계명대학교 비사연구기금으로 이루어졌음.
† 정 회 원 : 계명대학교 컴퓨터공학과 부교수
†† 준 회 원 : 경북대학교 전자전기컴퓨터학부 박사과정
††† 준 회 원 : 계명대학교 컴퓨터공학과 박사
논문접수 : 2009년 9월 9일
수 정 일 : 1차 2009년 10월 6일
심사완료 : 2009년 10월 7일

FlexRay의 기술발전에 기초가 되도록 하였다.

본 논문의 전체적인 구성은 다음과 같다. 2장에서는 최근의 AUTOSAR 기반 소프트웨어 기술의 동향과 AUTOSAR 및 FlexRay 표준에 대한 관련연구를 기술하였다. 3장에서는 이를 토대로 AUTOSAR 기반의 FlexRay 프로토콜 모듈의 전체적인 설계에 대해 기술하였으며, 4장에서는 설계를 바탕으로 구현 결과에 대한 시험 및 검증에 대하여 기술하였다. 마지막으로 5장에서는 본 연구의 결론과 향후 더 필요한 연구에 대하여 소개하였다.

2. 관련연구

2.1 AUTOSAR

AUTOSAR는 기술적으로 5가지의 궁극적인 목표[3]를 가지고 있다. 첫 번째 목표는 자동차 소프트웨어의 각 기능을 모듈화 하여 특정 ECU에 탑재될 소프트웨어를 개발할 때 필요한 모듈만을 선택적으로 탑재하여 구현할 수 있다. 두 번째 목표는 소프트웨어의 확장이 용이하도록 함으로써 소프트웨어의 업그레이드 및 수정을 쉽게 하는 것이다. 세 번째 목표는 ECU들에 탑재되는 소프트웨어의 기능별 분산이 자유롭게 이루어지도록 하는 것이다. 네 번째 목표는 소프트웨어의 재사용성을 높여서 동일한 회사의 여러 자동차에 탑재되는 소프트웨어 개발을 쉽게 할 수 있으며 품질과 신뢰성을 향상시킬 수 있도록 한다. 마지막으로 AUTOSAR는 표준화된 인터페이스를 통하여 여러 소프트웨어 공급업체들이 개발한 다양한 모듈이 잘 통합될 수 있도록 한다.

2.2 FlexRay

FlexRay는 자동차 네트워크에서 각 ECU를 연결하는 백본 네트워크에 사용된다[6]. FlexRay는 CAN과 LIN 프로토콜의 다음을 잇는 차세대 프로토콜로서 더욱 높은 신뢰성과 안전성이 특징이다. 특히 FlexRay는 이전의 다른 프로토콜보다 높은 데이터 전송률과 유연한 토폴로지, 결함방지 등의 장점을 지니고 있다[7]. 이러한 장점 때문에 FlexRay 프로토콜은 여러 자동차 전자제어 장치 중에서도 ABS를 비롯한 X-by-Wire 등 센서와 시스템 제어에서 주요한 통신 수단으로 주목받고 있다[8].

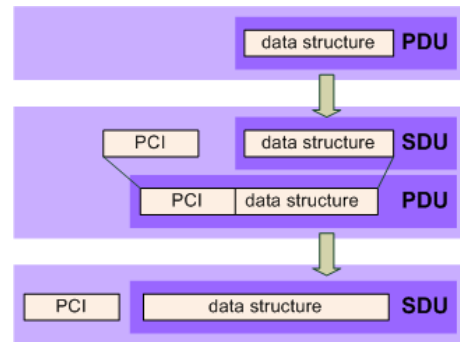
2.3 AUTOSAR 기반의 FlexRay 모듈

AUTOSAR는 자동차 내부 네트워크에 사용되는 FlexRay 프로토콜에 대한 표준 규격을 정의하고 있다. 기본 소프트웨어 계층 내에 구현되는 모듈들은 FlexRay 네트워크의 공통적인 인터페이스를 제공하여, 상위 어플리케이션에서 프로토콜과 메시지 특성들과 관계없이 인터페이스 사용이 가능하도록 해준다.

AUTOSAR 통신 모듈, 일반적인 네트워크 관리 인터페이스 모듈, 통신 진단 관리자 모듈 등은 모든 자동차 네트워크 시스템에 동일하게 구성되며 모든 ECU마다 존재한다. 일반적인 네트워크 관리 인터페이스 모듈은 시스템 서비스의 통신 관리자와 연결되며 디스패처 역할을 한다. 상태 관리자는 하드웨어 장치를 시작, 종료시킨다. PDU 라우터는 서로 다른 추상화된 통신 컨트롤러들과 상위 계층의 모듈들

사이에서 통신 경로를 제공한다.

(그림 1)에서는 AUTOSAR 기반의 계층화된 구조[9]에서 프로토콜 모듈들 간 인터페이스를 이용하여 PDU(Protocol Data Unit) 전달 흐름을 보여주고 있다. PDU는 PCI (Protocol Control Information)와 SDU(Service Data Unit)를 포함하고 있다. PCI는 하나의 모듈에서 다른 모듈에게 SDU를 전달하는데 필요하다.



(그림 1) 계층화된 구조에서 PDU의 흐름

3. AUTOSAR 기반 FlexRay 모듈 분석 및 설계

3.1 요구사항

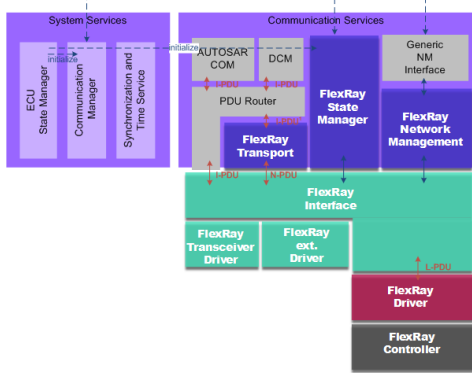
AUTOSAR에서는 차량 내 FlexRay 프로토콜 네트워크 구현을 위한 요구사항[10]을 다음과 같이 나타내고 있다.

- ① FlexRay Protocol 2.1 표준에 따라야 한다.
- ② FlexRay 네트워크 기능을 위한 다양한 인터페이스를 제공한다.
- ③ ECU 마다 최소 4개의 다른 FlexRay Controller를 지원한다.
- ④ 각각 10MBit/s의 속도를 가지는 최대 2개의 채널을 지원한다.
- ⑤ 다양한 FlexRay Cluster를 제공한다.
- ⑥ 동기 및 비동기 통신이 가능하도록 한다.
- ⑦ FlexRay 프레임에서 PDU 데이터들의 분리, 조합이 가능해야 한다.
- ⑧ 상위 계층에서는 PDU를 기반으로 제어되는 API들로 구성된 인터페이스를 사용할 수 있다.
- ⑨ 네트워크 관리를 위한 기본적인 기능을 제공한다.(예: 초기화, 종료, 통신 시작, 통신 시작 등)
- ⑩ FlexRay 통신의 상태와 버스의 에러 상태를 제공한다.

3.2 AUTOSAR 기반의 FlexRay 네트워크 시스템 분석

AUTOSAR 기반의 각 모듈들은 계층적 구조를 이루고 있다. FlexRay 네트워크 관련 기본 소프트웨어 모듈들도 계층적 구조를 이루고 있으며 (그림 2)에서 FlexRay 구조 및 인터페이스를 보여주고 있다.

AUTOSAR는 통신을 위하여 자동차 네트워크의 공통적인 기본 인터페이스를 제공한다. FlexRay 네트워크로 구성된 자동차 소프트웨어의 어플리케이션은 계층화된 구조의 모듈들을 이용하여 프로토콜과 메시지 속성을 자세히 알지 못하여도



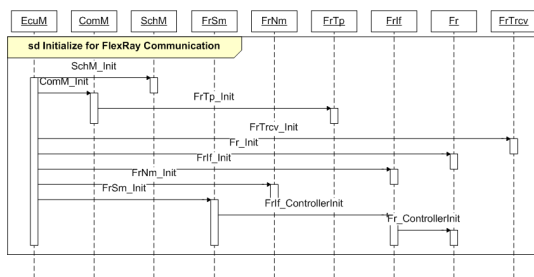
(그림 2) FlexRay 기본 소프트웨어 모듈 블록도

통신 기능을 사용할 수 있다. FlexRay State Manager[11]는 초기화 및 종료 등의 제어 기능을 통하여 통신 시스템을 운영 및 관리한다. 특히, FlexRay State Manager는 PDU를 보내거나 신호 타임아웃을 체크하는 등 통신상에서 필요한 다양한 옵션 기능들도 수행한다. FlexRay Transport Protocol[12] 모듈은 PDU 라우터와 연결되어 송수신되는 데이터를 분리 또는 조합하여 전달하는 기능을 수행한다.

통신 서비스 계층의 하위에는 하드웨어의 제어 기능과 연관이 있는 통신 하드웨어 추상화 계층이 있다. FlexRay Interface[13]는 FlexRay Driver[14]와 연결되어 드라이버의 여러 함수들을 사용할 수 있는 인터페이스를 제공한다. 이렇게 제공되는 인터페이스는 상위 계층에서 FlexRay Driver의 세부적인 내용을 알지 못하더라도 기능의 사용이 가능하게 한다.

물리적인 하드웨어를 직접 사용하는 FlexRay Driver는 최하위단인 통신 드라이버 계층의 모듈이다. FlexRay Driver는 하드웨어의 레지스트리 설정 및 초기화 등 물리적인 제어가 가능하도록 인터페이스를 제공해준다. 일반적으로 FlexRay Interface의 함수와 FlexRay Driver의 함수들은 일대일로 매치되어 인터페이스 함수를 사용한 하드웨어 제어가 가능하다.

AUTOSAR 기본 소프트웨어의 각 모듈 및 드라이버들은 AUTOSAR 서비스 계층의 ECU 관리자에 의해 초기화된다. 그리고 FlexRay 네트워크는 상태 체크 및 동작을 위해 FlexRay State Manager에 의해 시작된다. FlexRay 네트워크는 타임 트리거 방식의 네트워크이다. 네트워크의 동작은 기본 소프트웨어 스케줄러의 FIXED-CYCLIC 시간에 각 모듈의 메인 함수가 호출되면서 수행된다. 예를 들어 FlexRay Interface에서는 호출된 메인함수에서 글로벌 타임을 얻어오



(그림 3) FlexRay 통신 초기화

고, FlexRay Cluster에 해당하는 FlexRay Job List를 찾은 후, Absolute Timer Interrupt를 사용 가능하도록 만든다.

3.3 AUTOSAR 기반의 FlexRay 네트워크 동작 설계

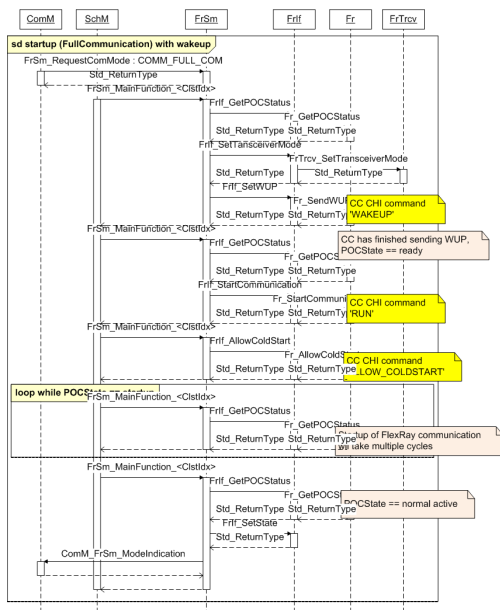
각 모듈들은 필요한 인터페이스를 제공하며 계층적 구조화를 이루어 서로 연결된다. 다음은 각 모듈들의 주요 시퀀스를 간략하게 보여주고 있다.

(그림 3)에서는 각 모듈들의 초기화 과정을 표현하고 있다. 각 모듈의 초기화는 서비스 계층의 ECU 매니저에 의해 수행된다. 초기화는 정상적인 하드웨어의 사용을 위해 반드시 선행적으로 이루어져야 한다. 그리고 연관되어 있는 모듈의 사용을 위해 순차적인 초기화가 이루어져야 한다. FlexRay Transport Protocol 모듈의 초기화 경우 ECU 관리자에 의해 초기화된 AUTOSAR 통신 관리자를 통하여 수행된다. 실질적인 FlexRay CC(Communication Controller) 하드웨어가 초기화되기 위해서는 FlexRay State Manager가 FlexRay Interface를 통하여 FlexRay Driver의 Fr_ControllerInit() 함수를 수행하여야 한다. POC:halt 상태의 FlexRay CC는 FlexRay Cluster 및 노드의 환경 매개변수 값들을 초기화하고 데이터 메시지 버퍼와 같은 송수신에 필요한 리소스들을 초기화하면서 POC:ready 상태로 변경된다.

(그림 4)에서는 초기화 과정 이후의 통신 시작 과정을 나타내고 있다. AUTOSAR 통신 관리자에서 COMM_FULL_COMMUNICATION 모드 설정으로 변경이 요청되면 FlexRay State Manager는 FlexRay 통신을 시작할 준비를 한다. FlexRay State Manager 동작은 메인함수에 의해 수행된다. 동작 수행 전 FlexRay Interface를 이용하여 FlexRay CC의 POC 상태를 알아본다. 만약 통신이 가능한 상태라면 FlexRay Driver에서 CHI(Controller Host Interface) 커맨드를 FlexRay CC에게 보내어 비로소 FlexRay 통신이 시작된다.

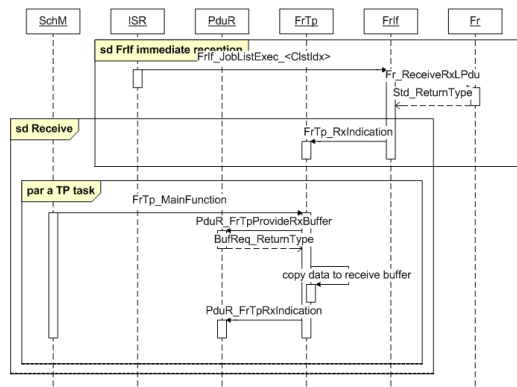
위의 시퀀스를 수행하는 동안 FlexRay CC의 POC 상태는 POC:startup을 거쳐 POC:normal active 상태로 변경된다. POC:normal active 상태는 정상적인 통신 상태로 다른 통신 모드를 요청하지 않는다면 FlexRay 프레임 데이터의 송수신이 가능하다. 데이터 송수신은 AUTOSAR의 계층화된 구조의 각 모듈들을 순차적으로 거치면서 수행된다. (그림 5)에서는 이러한 메시지 송수신 중 FlexRay Transport Protocol 모듈을 통한 데이터 송신 시퀀스를 간단하게 보여주고 있다.

FlexRay 네트워크의 데이터 송신은 상위 어플리케이션에서 보낸 메시지가 AUTOSAR 통신 모듈 중에서 PDU 라우터를 거쳐 전달되면서 이루어진다. PDU 라우터에서 메시지는 FlexRay Interface로 바로 전달될 수도 있고, FlexRay Transport Protocol 모듈을 거친 후 FlexRay Interface로 전달될 수도 있다. FlexRay Transport Protocol 모듈로 전달된 데이터는 메시지 길이 및 환경 변수 초기 값에 따라 여러 개의 세그먼트로 나뉘어져 첫번째 프레임과 연속 프레임으로 전달된다. 또한, 가능하다면 단일 프레임으로 전달되기도 한다. FlexRay Transport Protocol 모듈에서는 필요에 따라 응답 프레임을 전송하거나 데이터 전송에 실패한 경우 재시도 기능을 제공하기도 한다. 또, 흐름제어 기능을 제공

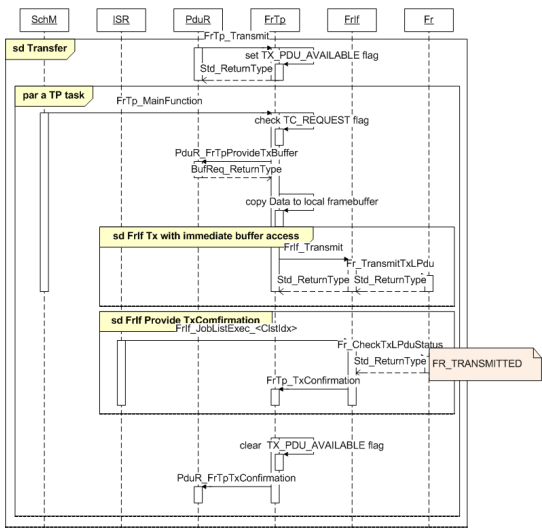


(그림 4) FlexRay 통신 시작 (Wakeup 포함)

모듈에서의 모든 동작 수행은 FIXED_CYCLE time에 수행되는 태스크에서 이루어진다. FlexRay Interface로부터 메시지가 수신되었음이 알려지면 수신된 데이터를 저장할 수 있는 버퍼를 제공받아 PDU 라우터에게 메시지가 수신되었음을 알린다.



(그림 6) FlexRay Transport Protocol 모듈을 통한 데이터 수신



(그림 5) FlexRay Transport Protocol 모듈을 통한 데이터 송신

하여 흐름 제어 데이터에 따라 알맞은 통신 기능을 수행한다. 메시지 수신은 OS의 인터럽트 서비스 루틴에 의해 호출되는 Job List Exec 함수에서 수행된다. Job List Exec 함수는 FlexRay Cluster 별로 정의되어 있다. Job List Exec 함수를 위한 인터럽트는 FlexRay CC의 Absolute Timer Interrupt에 의해 발생하며 본 논문에서는 OSEK 등의 OS가 없는 관계로 드라이버의 Absolute Timer Interrupt를 발생 가능하도록 함으로써 수행되도록 하였다. 메시지 수신 동작은 설정에 따라 버퍼에 저장 후 트리거 방식에 의해 전달될 수도 있고, 바로 전달될 수도 있다.

(그림 6)에서는 FlexRay Driver로부터 전달받은 데이터가 FlexRay Transport Protocol 모듈을 거쳐 PDU 라우터로 전달되는 과정을 보여주고 있다. FlexRay Transport Protocol

4. AUTOSAR 기반 FlexRay 모듈 구현 및 시험

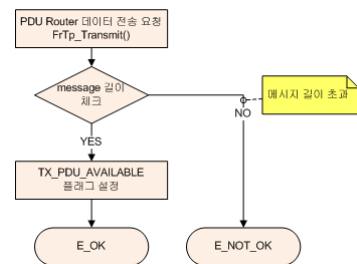
AUTOSAR 기반의 FlexRay 프로토콜 모듈의 구현은 Freescale Semiconductor사의 EVB9S12XF512E 보드[15]를 기반으로 개발하였다. EVB9S12XF512E 보드의 메인 칩셋인 MC9S12XF512는 FlexRay Interface가 내장되어 있는 Freescale의 16bit ECU 전용 칩셋이다.

4.1 FlexRay 프로토콜 주요 기능 함수

각 모듈들은 크게 초기화, 시작, 종료 등의 제어 함수들과 에러 및 상태를 체크하기 위한 상태 확인 함수, 그리고 FlexRay 네트워크 동작을 위한 메시지 송수신 함수로 구성된다.

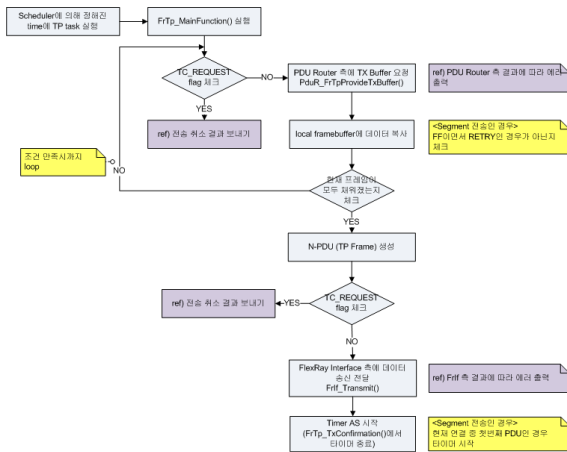
초기화 및 시작 과정은 필요 순서에 따라 ECU 관리자 및 AUTOSAR 통신 관리자, FlexRay State Manager에 의해 수행된다. FlexRay State Manager에서는 FlexRay Cluster를 초기화시키고 통신이 수행될 준비를 한다. FlexRay CC가 초기화되고 정상적으로 통신 준비가 완료되었다면 AUTOSAR 통신 관리자에서 FlexRay State Manager에게 FULL COMMUNICATION 통신 모드를 요청하면서 통신이 시작된다.

FlexRay 트랜스포트 모듈은 PDU 라우터와 FlexRay

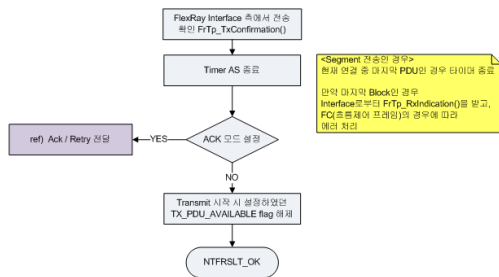


(그림 7) FlexRay 트랜스포트에게 데이터 전송 요청

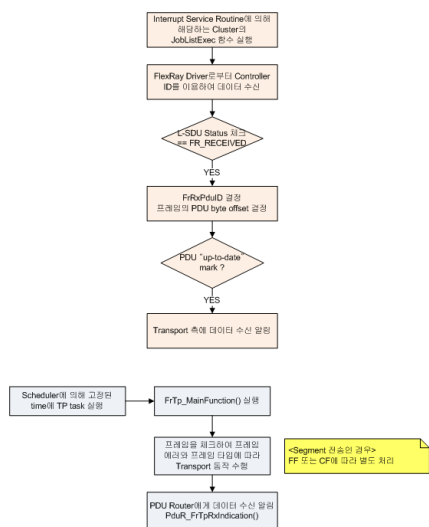
Interface 사이에서 데이터가 전달될 수 있도록 메시지를 분해하거나, 조합하고 흐름을 제어하는 등 다양한 기능을 수행한다. 대표적인 메시지 송신 과정을 살펴보면 (그림 7)부터 (그림 9)까지의 과정과 같다. (그림 7)에서 보는 바와 같이 PDU 라우터에서 트랜스포트에게 데이터 송신을 요청하면 일단 플래그를 설정한 후 리턴한다.



(그림 8) FlexRay Interface에게 데이터 전송 요청



(그림 9) 완료된 데이터 전송에 대한 확인 전달



(그림 10) FlexRay 네트워크에서의 데이터 수신 흐름도

(그림 8)에서처럼 메인 함수에서 적절한 프레임 생성하여 FlexRay Interface 측에 전달하고, (그림 9)에서는 전송 완료 후 FlexRay Interface로부터 데이터 전송 확인이 전달되고, 정상 완료되었다면 결과를 PDU 라우터 측에 확인 전달한다. 데이터 수신시 경우 FlexRay Interface에서는 각 Cluster마다 할당된 JobListExec 함수에서 수행된다. FrIf_JobListExec_<Cluster ID> 함수는 연결된 FlexRay CC의 Absolute Timer Interrupt에 의해 발생한다. FlexRay Driver로부터 L-PDU 데이터를 수신하면 FlexRay 트랜스포트 모듈에게 알리고 데이터를 가져갈 수 있도록 한다. FlexRay 트랜스포트 모듈에서는 데이터 송신과 마찬가지로 스케줄러에 의한 메인 함수에서 프레임 데이터를 확인하고 필요하다면 데이터를 조합하여 PDU 라우터 측에 데이터를 전달한다. (그림 10)에서는 이러한 과정을 흐름도로 나타내었다.

4.2 시험 환경 및 검증 결과

구현에 대한 시험은 개발보드인 2개의 ENV9S12XF512E 보드를 각각 FlexRay A, B 두 개의 채널로 연결하여 이루어졌다. 보드의 연결은 (그림 11)과 같다.

각 보드는 하나의 ECU와 같이 동작하며, FlexRay 네트워크로 연결되어 있다. 두 개의 보드를 연결하는 선이 FlexRay를 위한 네트워크 연결선이며 A, B 두개의 채널로 연결되어 있다. FlexRay를 통하여 송수신된 상태는 보드의 LED의 점멸로 표시된다. 검증을 위하여 각 모듈은 다음과 같이 동작되도록 하였다.

<Node1에서의 LED 동작>

- Slot 1에서의 송신 시 D23 LED ON
- Slot 4에서의 수신 시 D25 LED ON
- Slot 62에서의 수신 시 D28 LED ON

<Node2에서의 LED 동작>

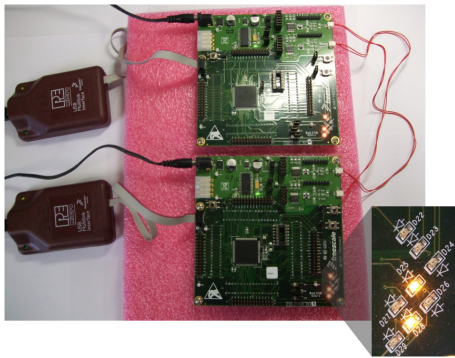
- Slot 1에서의 수신 시 D23 LED ON
- Slot 4에서의 송신 시 D25 LED ON
- Slot 62에서의 송신 시 D28 LED ON

FlexRay 데이터의 송수신과 관계없이 각 모듈이 정상적으로 동작하고 있는지 여부를 확인하기 위하여 D22, D27, D29의 별도 LED가 지속적으로 점멸되도록 하였다.

이 시험을 통하여 AUTOSAR 기반으로 구현된 FlexRay 데이터 송수신을 LED 점멸을 통해 확인할 수 있었으며, 계층화된 단계별 모듈 동작 역시 확인 가능하였다.

AUTOSAR 모듈의 검증 결과는 AUTOSAR 컨소시엄의 회원사로 등록되어 있는 업체들을 기준으로 정해진 인증센터에 의해 성능에 대한 검증 및 규격에 대한 검증이 필요하다. 하지만 본 논문에서 구현된 내용은 아직 OS, RTE 등의 AUTOSAR 전체적인 구조에 걸친 시험 결과가 없어 정식 과정을 거치는 검증이 어렵다. 이에 동작에 대한 검증 결과는 기존의 EVB9S12XF512E 보드에 구현되어 있는 Freescale사에서 제공되는 FlexRay Unified Driver의 동작과 비교하여 모듈들의 기능을 확인하였다.

구현된 결과의 검증은 객관적인 확인을 위하여 앞서 기술된 AUTOSAR가 제시하고 있는 FlexRay 네트워크의 기본적인 요구사항들에 대한 구현 결과 확인으로 체크하였다.



(그림 11) 2개의 EVB9S12XF512E 보드 연결 (Stand-alone mode)

5. 결 론

본 논문에서는 최근 자동차 임베디드 시스템 분야에서 크게 대두되고 있는 AUTOSAR 표준 플랫폼을 기반으로 하여 FlexRay 네트워크 모듈을 구현한 결과를 제시하였다. AUTOSAR 규격에서 제시하고 있는 State Manager, Transport, Interface, Driver, Transceiver Driver 등의 모듈을 구현함으로써 계층화된 소프트웨어 구조에 따른 모듈별 관계를 입증하였다. 또한 구현된 결과를 Freescale MCS9S12XF512 마이크로컨트롤러에 포팅하여 FlexRay 프로토콜 네트워크의 동작을 확인하였다.

AUTOSAR는 현 단계의 표준도 상용 자동차에서 적용되고 있으나 아직 그 규격이 안정화 단계까지 진행되지는 못하였다. AUTOSAR 컨소시엄은 다양한 구현 및 검증 단계를 거쳐 2009년까지 Full Release 4.0 규격 발표를 목표로 하고 있다. 향후에는 본 논문에서 연구한 결과를 바탕으로 OSEK등의 OS 및 RTE 등을 포함한 AUTOSAR 전체 규격을 구성하고 자동차 소프트웨어의 완성도를 높일 것이다. 이로써 AUTOSAR 표준에서의 FlexRay 네트워크 규격에 대한 학술적 연구의 기초가 될 것이다.

참 고 문 헌

[1] Nilsson Dennis K., Phung Phu H., Larson Ulf E., "Vehicle ECU classification based on safety-security characteristics," Road Transport Information and Control - RTIC 2008 and ITS United Kingdom Members' Conference, IET, May 2008.

[2] Schreiner D., Goschka K.M., "A Component Model for the AUTOSAR Virtual Function Bus," Computer Software and Applications Conference, 31st Annual International, July 2007.

[3] Automotive Open System Architecture, "AUTOSAR," available at URL: <http://www.autosar.org>.

[4] AUTOSAR GbR, AUTOSAR Technical Overview V2.2.1 R3.0 Rev 0001, AUTOSAR, 02. 2008.

[5] FlexRay consortium, "FlexRay - The communication system for advanced automotive control applications," available at URL: <http://www.flexray.com>.

[6] Makowitz R., Temple C., "Flexray - A communication network for automotive control systems," Factory

Communication Systems, 2006 IEEE International Workshop on, June, 2006.

[7] FlexRay Communication System Protocol Specification Version 2.1 Revision A, 2005.

[8] 김병우, "미래형 자동차의 정보 네트워크 기술," 정보처리학회지 제14권 제1호, 2007. 1.

[9] AUTOSAR GbR, AUTOSAR Layered Software Architecture V2.2.1 R3.0 Rev 0001, AUTOSAR, 02. 2008.

[10] AUTOSAR GbR, Requirements on FlexRay V2.0.4 R3.0 Rev 0001, AUTOSAR, 10. 2007.

[11] AUTOSAR GbR, Specification of FlexRay State Manager V1.0.2 R3.0 Rev 0002, AUTOSAR 02.2008.

[12] AUTOSAR GbR, Specification of FlexRay Transport Layer V2.2.1 R3.0 Rev 0001, AUTOSAR, 02. 2008.

[13] AUTOSAR GbR, Specification of FlexRay Interface V3.0.2 R3.0 Rev 0002, AUTOSAR, 02. 2008.

[14] AUTOSAR GbR, Specification of FlexRay Driver V2.2.1 R3.0 Rev 0001, AUTOSAR, 12. 2008.

[15] Freescale Semiconductor, EVB9S12XF512E User Manual Rev. 0, Freescale Semiconductor, 06. 2007.



주 흥 택

e-mail : juht@kmu.ac.kr
 1989년 한국과학기술원 전산학(학사)
 1991년 포항공과대학교 컴퓨터공학과 (공학석사)
 2002년 포항공과대학교 컴퓨터공학과 (공학박사)

1991년 9월~1997년 2월 대우통신, 종합연구소, 선임연구원
 2009년 10월~현 재 계명대학교 컴퓨터공학과 부교수
 관심분야 : Web-based Network Management, Network Monitoring, Mobile Device Management



정 설 영

e-mail : snowflower@ee.knu.ac.kr
 2001년 동아대학교 컴퓨터공학과(학사)
 2009년 계명대학교 (전산교육 석사)
 2009년 경북대학교 전자전기컴퓨터학부(박사과정)
 2001년~2004년 (주)유진로봇 Researcher

2005년~2008년 (주)맥산 Assistant Researcher
 관심분야 : 자동차네트워크, 센서네트워크, 홈네트워크, RTOS 등



이 명 속

e-mail : mslee@kmu.ac.kr
 2001년 계명대학교 컴퓨터공학과(학사)
 2003년 계명대학교 컴퓨터공학과 (공학석사)
 2009년 계명대학교 컴퓨터공학과 (공학박사)
 관심분야 : XML 기반의 네트워크 관리, 네트워크 모니터링