

# OSEK/VDX COM을 위한 내부 통신 메커니즘 설계

이우영\*, 임성락\*  
 \*호서대학교 컴퓨터공학과  
 e-mail : wmk2@naver.com

## Design of Internal Communication Mechanism for OSEK/VDX COM

Woo-Young Lee\*, Seong-Rak Rim\*  
 \*Dept. of Computer Engineering, Hoseo University

### 요 약

OSEK/VDX COM은 자동차 ECU의 통신을 위해 OSEK/VDX에서 제안한 통신 규약이다. 본 논문에서는 OSEK/VDX COM의 내부 통신 메커니즘을 설계한다. 설계한 메커니즘의 타당성을 검증하기 위해 ARM7 프로세서가 장착된 임베디드 개발 보드 EZ-AT7를 이용하여 ECU 내부의 태스크간의 통신의 정상적인 동작을 확인 하였다.

### 1. 서론

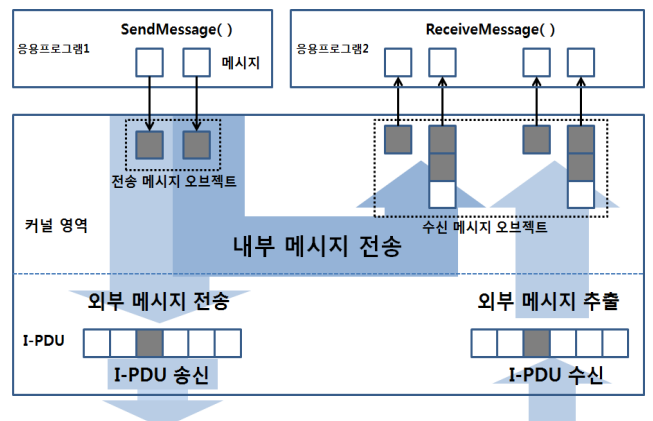
OSEK/VDX는 자동차 전자 제어 장치(ECU:Electronic Control Unit)를 위한 산업계 표준을 제시하고자 구성된 협의체로서 운영체제(OSEK/VDX OS), 통신(OSEK/VDX COM), 네트워크 관리(OSEK/VDX NM) 그리고 구현 언어(OIL) 규약을 제안하고 있다.[1] OSEK/VDX COM은 자동차용 응용프로그램을 위한 통신 환경으로 이식성을 높이기 위해 일반적인 통신 인터페이스와 ECU 내부 및 외부 통신 특성을 정의 하고 있다.[2]

본 논문에서는 OSEK/VDX COM의 내부 및 외부 통신을 소개하고, 그 중에서 내부 통신 메커니즘을 설계한다. 설계한 메커니즘을 검증하기 위하여 ARM7 프로세서가 장착된 임베디드 개발 보드 EZ-AT7[3]을 통하여 동작상태를 확인한다.

#### 1.1 OSEK/VDX COM 내부 및 외부 통신

OSEK/VDX COM은 (그림 1)과 같이 내부 및 외부 통신을 제시하고 있다.

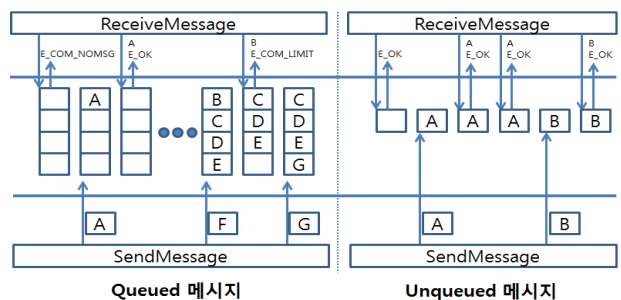
(그림 1)에서 메시지는 시스템 생성 전에 정적으로 정의되고, 응용 프로그램의 태스크와 인터럽트 서비스 루틴에서만 전송이나 수신이 가능하다. 메시지는 응용 프로그램에서 SendMessage()를 통해 전송 메시지 오브젝트로 전송하고, ReciveMessage()를 통해 수신 메시지 오브젝트로부터 수신한다. 내부 통신의 경우 전송 오브젝트의 메시지를 바로 수신 오브젝트로 전달한다. 외부 통신은 I-PDU를 통해 다른 ECU의 I-PDU로 송신하고, 수신한 I-PDU에서 수신 메시지 오브젝트로 전달한다.



(그림 1) OSEK/VDX COM 내부 및 외부 통신

#### 1.2 Queued & Unqueued 메시지

OSEK/VDX COM은 수신 메시지 오브젝트의 큐의 자료구조에 따라 (그림 2)와 같이 Queued 메시지와 Unqueued 메시지 2가지 방식을 제공한다.



(그림 2) Queued/Unqueued 메시지 동작 방식

(그림 2)에서 Queued 메시지는 FIFO 형태의 환영 큐를 사용한다. SendMessage()는 큐에 비어있는 공간이 있다면 메시지를 저장한다. ReceiveMessage()는 큐에 메시지가 있으면 메시지를 가져오고, 큐에서는 삭제된다.

Unqueued 메시지는 오직 하나의 메시지만 저장한다. SendMessage()는 큐 공간을 하나만 가지기 때문에 메시지를 매번 저장하여 기존의 메시지는 사라진다. ReceiveMessage()는 큐로부터 메시지를 가져오고, 큐에는 메시지가 남아있다.[1]

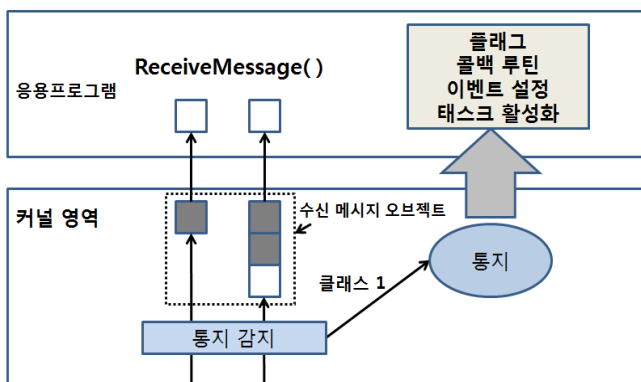
### 1.3 통지 클래스

OSEK/VDX COM에서는 전송이나 수신에 최종 상태를 확인하기 위해서 <표 1>과 같이 4가지의 통지 클래스를 지원한다.

<표 1> 통지 클래스

통지 클래스	해당 통신
클래스1 : 메시지 수신	내부 통신, 외부 통신
클래스2 : 메시지 전송	외부 통신
클래스3 : 메시지 수신 에러	외부 통신
클래스4 : 메시지 전송 에러	외부 통신

통지 클래스1은 메시지가 수신 오브젝트에 저장될 때 통지하고, 통지 클래스2는 메시지를 포함한 I-PDU를 성공적으로 전송 후에 통지한다. 통지 클래스3은 메시지 수신 에러가 감지된 경우 통지하고, 클래스4는 메시지 전송 에러가 감지된 경우 통지를 한다. <표 1>에서 모든 통지 클래스는 외부 통신을 지원하지만 내부통신은 오직 통지 클래스1만 지원함을 알 수 있다.



(그림 3) 통지 클래스1 동작

통지 클래스1은 (그림 4)와 같이 메시지가 수신 메시지 오브젝트에 저장될 때 통지를 발생시킨다. 통지 방법에는 4가지(플래그 설정, 콜백 루틴 호출, 이벤트 설정, 태스크 활성화)가 있으며 수신 메시지 오브젝트에 설정된 방법으로 응용 프로그램에 알린다.

## 2. 설계

### 2.1 메시지 구조체

OSEK/VDX COM의 내부 통신을 위하여 <표 2>과 같이 구조체를 정의한다.

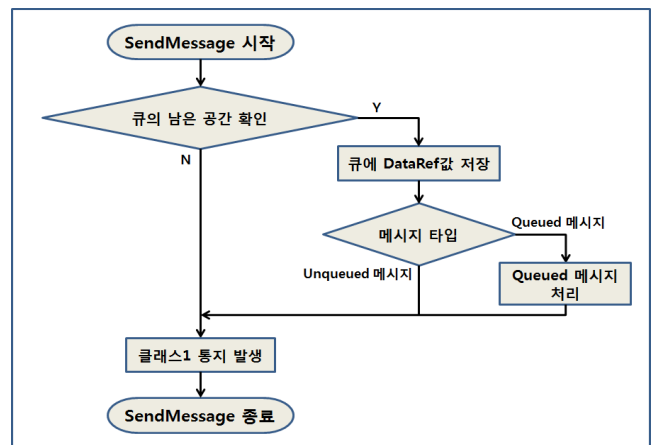
<표 2> 메시지 구조체 속성

데이터	이름	속성
u8	msg_type	메시지 종류
u8	queued_type	큐 타입
u8	queued_size	큐 크기 (unqueued 는 1)
u32 *	msg_start_addr	메시지 저장 공간의 시작 주소
u32 *	msg_send_addr	메시지 저장 공간의 송신 주소
u32 *	msg_recv_addr	메시지 저장 공간의 수신 주소
u8	msg_count	저장된 메시지 수
u8	rx_notification	통지 클래스 1

메시지 종류에는 내부전송, 외부전송, 송신, 수신 4가지가 있다. 이는 동시에 사용될 수 있기 때문에 msg\_type으로 정의하고, 비트단위로 구분한다. 수신 메시지 오브젝트의 큐 타입은 queued\_type에 정의한다. Queued 메시지의 환영 큐를 제어하기 위해 queued\_size, msg\_start\_addr, msg\_send\_addr, msg\_recv\_addr, msg\_count를 정의한다. 통지 클래스1을 위하여 rx\_notification을 정의한다.

### 2.2 SendMessage(Message, DataRef)

SendMessage()는 (그림 5)와 같이 DataRef를 참조해 호출한 응용 프로그램으로부터 데이터를 가져오고 Message에 의해 지정된 수신 메시지 오브젝트에 저장한다.



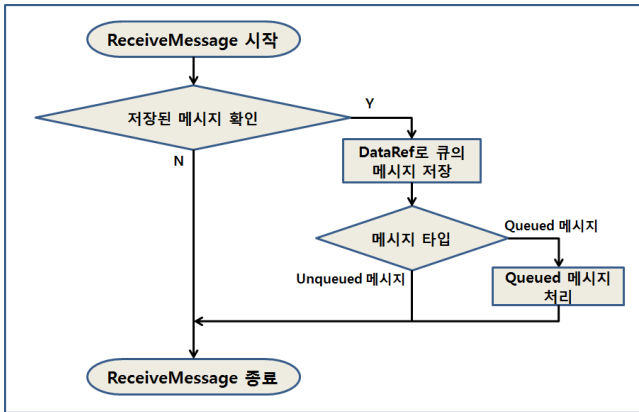
(그림 5) SendMessage() 순서도

(그림 5)에서 SendMessage()는 큐에 남은 저장 공간이 있는지 확인하고, 남은 저장 공간이 있을 경우 msg\_send\_addr의 위치에 DataRef 값을 저장한다. 저장 후 Queued 메시지는 msg\_count 값을 증가시키고, msg\_send\_addr를 다음 저장 위치를 가리킨다. 모든 작업

완료 후 클래스1 통지를 발생한다. 통지는 rx\_notification 값을 보고 플래그 설정, 콜백 루틴 호출, 이벤트 설정, 태스크 활성화 중 하나의 방법을 처리한다.

### 2.3 ReciveMessage(Message, DataRef)

ReciveMessage() (그림 6)과 같이 Message에 의해 지정된 수신 메시지 오브젝트로부터 데이터를 가져와서 DataRef를 참조하여 응용 프로그램에 저장한다.



(그림 6) ReciveMessage() 순서도

(그림 6)에서 ReciveMessage()는 큐에 저장된 메시지가 있는지 확인하고, 메시지가 있을 경우 msg\_rcv\_addr의 위치의 데이터를 DataRef에 저장한다. 저장 후 Queued 메시지는 msg\_count 값을 감소 시키고, msg\_rcv\_addr를 다음 수신 위치를 가리킨다.

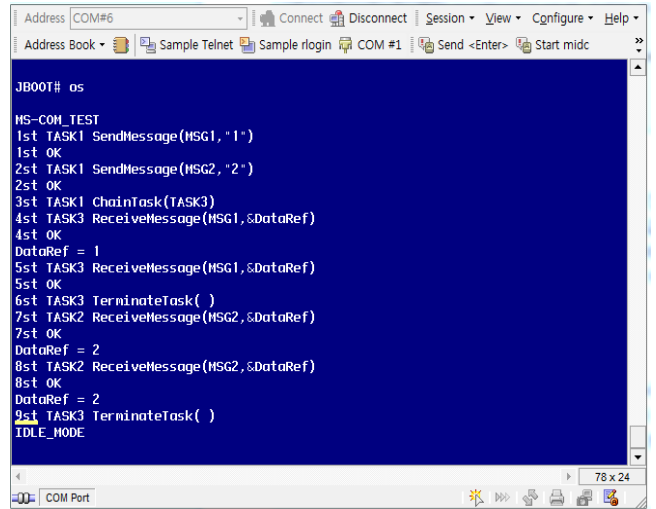
### 3. 실험

설계한 루틴의 타당성을 검증하기 위해서 OSEK/VDX에서 제공한 테스트 시퀀스를 참조하여 <표 3>와 같이 만들어 ARM7 프로세서가 장착된 임베디드 개발 보드 EZ-AT7을 통해 테스트 한다.[4]

<표 3> 테스트 시퀀스

순서	태스크	서비스 루틴	리턴 상태
1	TASK1	SendMessage(MSG1, "1")	E_OK
2	TASK1	SendMessage(MSG2, "2")	E_OK
3	TASK1	ChainTask(TASK3)	E_OK
4	TASK3	ReceiveMessage(MSG1, &DataRef)	E_OK DataRef = 1
5	TASK3	ReceiveMessage(MSG1, &DataRef)	E_COM_NOMSG
6	TASK3	TerminateTask()	
7	TASK2	ReceiveMessage(MSG2, &DataRef)	E_OK DataRef = 2
8	TASK2	ReceiveMessage(MSG2, &DataRef)	E_OK DataRef = 2
9	TASK2	TerminateTask()	

태스크 이름과 같은 우선순위를 갖는 태스크를 3개 만들고, Queued 메시지 타입의 MSG1 메시지와 Unqueued 메시지 타입의 MSG2를 가지고 테스트 한다. MSG1 메시지는 태스크 활성화 통지 메커니즘을 설정하여 TASK2를 활성화 한다.



(그림 7) 테스트 결과

(그림 7)은 EZ-AT7을 통해 <표 3>의 시퀀스를 수행한 결과이다. TASK1에서 MSG1과 MSG2로 메시지를 전송하고, TASK3을 활성화 한다. TASK3에서는 MSG1의 메시지를 수신하여 Queued 메시지 타입의 내부 전송과 통지 메커니즘의 정상적인 동작을 확인하고, TASK2에서 MSG2의 메시지를 수신하여 Unqueued 메시지 타입의 내부 전송의 정상적인 동작을 확인 하였다.

### 4. 결과

본 논문에서는 OSEK/VDX COM에서 제안한 요구사항을 고려하여 내부 통신의 메커니즘을 설계했다. ARM7 프로세서가 장착된 임베디드 개발 보드 EZ-AT7를 통해 구현한 SendMessage()와 ReceiveMessage()의 타당성을 검증하여 내부 통신의 정상적인 동작을 확인 하였다.

### 참고문헌

[1] 임성락, "OSEK COM의 내부통신을 위한 메시지 송수신 메커니즘 설계", 한국산학기술학회 논문집, 5월, 2011.

[2] OSEK/VDX Communication Specification 3.0.3, (<http://www.osek-vdx.org>) 2월, 2005.

[3] <http://forum.falinux.com>

[4] OSEK/VDX Communication Test Plan 2.0, (<http://www.osek-vdx.org>) 4월, 1999.