

TMO - eCos 기반의 무인자동차 제어 프레임워크와 응용

김현주⁰ 서한석 김성진 김정국

한국외국어대학교 대학원 컴퓨터 및 정보통신학과
{kimhj⁰, Hanseok, sungjin04, jgkim}@hufs.ac.kr.

Automobile Control Framework and application using eCos-based TMO Engines

Hyun - Joo, kim⁰ Han-Seok Seo, Seng -Jin kim Jung-Guk, Kim
Dept. of Computer and Information Communication Engineering,
Hankuk University of Foreign Studies

요 약

무인 자동차 시스템과 같은 실시간 제어 환경에서는 각종 센서의 상황에 대한 주기적인 폴링, 실시간 스케줄링, 병행 메소드의 지원, 실시간 병행 접근 제어 등과 같은 환경이 요구된다.

본 논문에서는 micro 내장형 운영체제의 실시간 객체 엔진으로 개발한 TMO-eCos를 기반으로, TMO를 이용한 무인 자동차 제어 프레임워크와 이를 활용한 응용 모델에 대해 기술한다. TMO 모델을 이용한 무인 자동차 제어 프레임워크는 실시간 제어 시스템 개발을 위한 객체 기반의 규격적 환경을 제공하여, 최근 국내외적으로 많은 연구가 진행되고 있는 지능형 실시간 로봇 제어 소프트웨어의 기본 플랫폼으로 활용될 수 있을 것이다.

1. 서론

TMO (Time-triggered Message-triggered Object) 모델은 규격적 실시간 프로그래밍 설계를 위해 제안된 모델로 객체 지향 프로그래밍, 시간과 메시지에 의해 구동하는 동적 실시간 스케드의 객체 멤버화, 데드라인 스케줄링 및 분산 IPC 등을 통합적으로 제공하는 모델로 정시 보장성을 위한 TMO 어플리케이션들은 여러 운영체제에 미들웨어와 커널형태로 개발되었다. 그러나 많은 내장형 장치들로 구성되는 실시간 네트워크 제어를 위한 TMO 객체 지원 커널에는 스케줄링의 정확성 이외에도 작은 크기가 요구된다. TMO-eCos[2]는 내장형 오픈소스 마이크로 운영체제인 eCos를 기반으로 개발된 TMO 지원 임베디드 커널로 로봇 제어 시스템과 같은 제어 시스템에 적용될 때 모듈화 구성이나 용량면에서 큰 이점을 갖는다.

본 논문은 TMO-eCos를 기반으로 무인자동차 제어 모델링에 을 위한 프레임워크를 제시하고 이를 활용한 응용 모델을 개발함으로써, 실시간 임베디드 소프트웨어 분야에서 TMO의 편이성 및 실시간성을 검증한다.

2. TMO 모델

TMO (Time-triggered Message-triggered Object)[1] 실시간 프로그래밍, 분산 시스템 프로그래밍, 병행 프로그래밍, 객체 지향 프로그래밍의 모든 장점을 통합한, 분산 실시간 객체 모델로, '90년대 초반부터 세계적으로 주목 받고 있는 새로운 프로그래밍 패러다임이다. 일반적으로 복수 개의 TMO로 설계된 시스템은 분산 환경에서 주어진 시간 조건에 의해 수행된다. TMO-eCos는 이러한 TMO로 구성된 프로그램을 단일 또는 복수 개의 embedded 플랫폼에서 수행하기 위해 개발된 엔진이다.¹

¹ 본 논문은 정보통신부 ITRC의 지원에 의한 것이다.

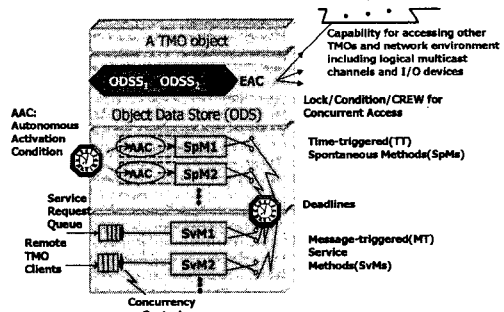


그림 1. TMO의 구조[1]

TMO는 다음과 같은 특징을 갖는다.

- 일반적인 객체의 멤버 형태 이외에 새로운 두 가지 형태의 실시간 멤버 스케드를 가질 수 있는 동적 객체이다.
- TMO 객체 내의 자료 멤버를 공유하는 멤버 스케드는 주어진 시간 조건에 의해 동작하는 SpM (Time-triggered Method: Spontaneous Method)과, 분산 및 국부 IPC 메시지의 수신에 의해 동작하는 SvM (Message-triggered Method: Service Method)의 두 가지 형태가 있다.
- SpM은 주기에 의해 구동되고 구동 시작 후에는 주어진 finish deadline에 의해 scheduling 된다.
- SvM은 IPC event 메시지의 수신에 의해 구동되고 구동인 시작되면 주어진 deadline에 의해 scheduling 된다.
- SpM과 SvM들이 객체 내의 ODS (Object Data Store)를 동시에 접근할 때의 상호배제(mutual exclusion)를 위해

기본적인 LOCK 구조체와 동기화 도구를 포함하는 CREW (Concurrent-Read Exclusive-Write) 모니터를 사용한다.

- TMO는 경성 또는 연성 실시간 응용뿐만 아니라 일반적인 분산 병행 프로그램 응용에도 사용할 수 있는 객체 모델이다.

경성 실시간 응용의 경우, TMO는 SpM에 의한 설계 시 시간 보장 개념을 제공한다.

2.1. TMO-eCos

TMO-eCos는 TMO 기반의 분산 실시간 객체의 실행을 위해 다음과 같은 기능을 제공한다.

- CPU의 성능에 따라 최소 30us에서 10ms까지의 정밀도로 time-triggered 스레드와 message-triggered 스레드의 데드라인 구동 실시간 스케줄링을 제공
- Time-triggered 실시간 스레드의 정시 구동(on-time activation)을 제공
- 논리적인 멀티태스킹 IPC 서브시스템을 제공하며, 원격 및 로컬 메시지를 통한 message-triggered 실시간 스레드 구동 기능을 제공
- 분산 노드간의 클럭 동기화 제공
- TMOSL(TMO Support Library)를 이용한 TMO 기반의 프로그래밍 방법을 제공

다음 그림 2는 TMO-eCos 구조이다.

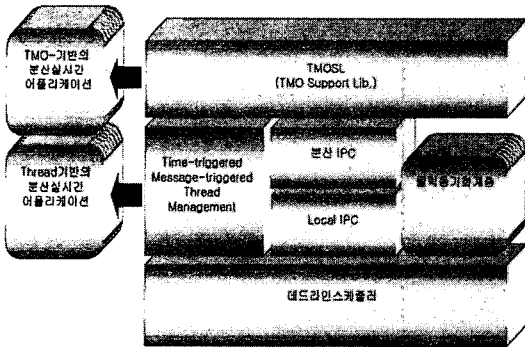


그림 2. TMO-eCos 구조

TMO-eCos 커널을 이용하여 분산 실시간 어플리케이션을 개발하는 방법에는 2 가지가 있다.

- 스레드 기반의 방식으로 TMO-eCos가 제공하는 기본적인 API를 이용
- TMO 기반 방식으로 TMOSL를 이용

3. TMO-eCos 기반의 무인자동차 제어 프레임워크

TMO-eCos를 이용한 로봇 제어 시스템 프레임워크는 다양한 로봇의 하드웨어 플랫폼에서 수정 없이 재사용이 가능하며 TMO 객체 구조와 스레드 방식에서의 TMO 의해 컴포넌트 구성 및 재구성성이 용이하다. 또한 실 세계의 객체 모델링을 쉽게 프로그래밍에 적용할 수 있어 개발 소프트웨어 및 객체, 컴포넌트의 통합이 대단히 쉽다. 스레드 기반의 TMO-eCos를 이용한 무인 자동차 제어 프레임워크의 특징은 다음과 같다.

- Time_triggered(SpM)들은 센서 등의 장치 종류를 Polling하거나, 구동장치를 주기적으로 제어하는데 사용되며, 이러한 SpM들은 적절한 주기(period)와 장치 특성에 따른 자료 수집 기간을 데드라인 파라미터로 설정함으로써 일반화 될 수 있다. 이들 메소드들은 ODS내의 각종 센서 상태, 구동 장치 상태들의 실시간 자료들을 주기적으로 업데이트하는 역할을 한다.
- Message_triggered(SvM)들은 각 구동 장치의 제어에 의한 기간 내 구동, 외부 입력에 대한 처리 등을 구현하는 메소드들로 활용된다. 이러한 SvM들은 처리 기간인 데드라인과 메시지 송수신 채널, 이벤트 수신시의 동작 등으로 구격화 된다.
- Controller_SpM은 구동, 입력 이벤트, 센서 장치를 담당하는 SpM과 SvM들 이외에 ODS의 로봇 상황을 전반적으로 점검하여 각 구동 SvM을 동작시키는 종합 판단 로직을 갖는 중추 스레드의 역할을 한다.

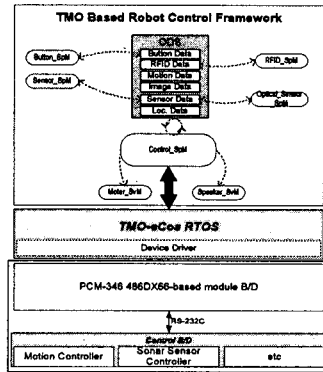


그림 3. TMO-eCos기반의 제어 프레임워크

4. 무인 자동차 제어 프레임워크의 응용

본 논문의 예시 응용은 제 2 회 임베디드 소프트웨어 공모 대전의 지정과제인 무인 자동차 로봇에 의한 운전 면허 시험을 대상으로 TMO-eCos를 탑재 재구현한 것이다.

무인 자동차 로봇의 주요 하드웨어/소프트웨어 사양은 다음과 같다.

- 구동부: 8 비트 프로세서 탑재 모터 구동 및 16 개의 광조도 센서 작동
- 제어부: 메인 로직이 있는 보드로 PCM-346 486DX66-base module 보드로 본 연구에서 개발된 제어 프레임워크를 탑재

TMO 기반 무인 자동차 제어 응용프로그램은 TMO 객체 실행을 위한 프레임워크 상의 광센서 입력을 위한 SpM, 로봇 구동을 위한 SvM, 종합적 구동을 지시하는 제어 SpM 들의 컴포넌트 메소드들로구성된다. 그림 4 는 TMO-eCos를 기반으로 한 무인 자동차 제어 프레임워크 응용인 운전 면허 시험의 구조도이다. 다음은 각 메소드 구성요소에 대한 설명이다.

- 광센서 SpM : 10 milli-sec 간격으로 앞뒤 16 개 광센서 입력을 받아 ODS에 기록한다.

- 구동부 SvM : 제어 SpM으로부터 로봇 조향, 진행 등의 명령을 수신하여 로봇 구동부에 전달한다.
- 제어 SpM : 10 milli-sec 간격으로 ODS의 광센서 상태와 먼저 시험 주행 상태를 스캔하여 다음 구동을 결정하고 이를 구동부 SvM에 지시한다.

- T 코스: T 코스의 경우 우로 90 도 회전 후, 뒤로 전진하며 정지선을 만나게 되면 정지 한다. 정지 한 후 다시 코스를 빠져 나가게 된다.
- 돌발 코스: 광 센서에 의해 돌발 코스가 인식되면 급정거를 수행한다. 정지 3 초간 대기 후 다시 주행을 시작한다.
- 주차 코스: 좌로 90 도 회전 후 정지선을 만나게 되면 다시 우로 회전하여 3 초간 대기한다. 정지 후 코스를 빠져 나간다.
- S 코스: 코스의 굴곡선을 따라가며 조향 계수(보상값)를 제어 된다. 조향 알고리즘은 후진 시에도 그대로 적용된다.

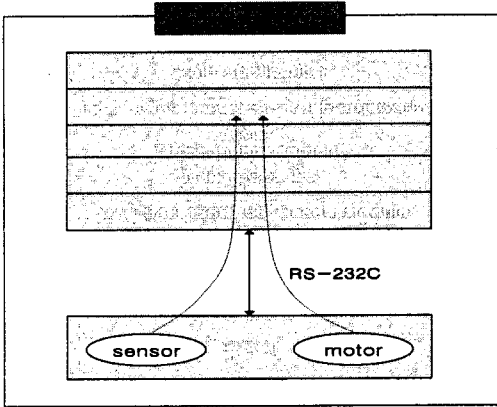


그림 4. 무인 자동차 제어 프레임워크의 응용구조도

제어 SpM의 로직은 ODS의 스캔을 바탕으로 한, 조향 알고리즘 및 각 코스 주행 알고리즘으로 구성된다.

4.1 조향알고리즘

조향 알고리즘은 무인 자동차가 기본적으로 전진하기 위한 알고리즘이다. 무인 자동차가 기동하면 바퀴 자체의 비원형성 및 축의 틀어짐, 등 여러 가지 원인에 의해 정확하게 직진을 하지 못하게 된다. 즉 하드웨어적으로 완전 무결한 시스템을 만드는 것은 어렵기 때문에 조향 알고리즘은 필수적이다. 본 개발에서 활용하는 광조도 센서는 앞뒤에 8 개씩 총 16 개의 센서를 통해서 센싱을 하게 된다. 바깥쪽 광조도 센서의 값을 통해서 현재의 조향 상태를 파악하고 정상적으로 전진할 수 있도록 한다. 다음은 조향 알고리즘을 적용한 간단한 예시코드이다.

```

if (gear->getStatus() == forward) {
    if ((sensor.front == 0xc0) || (sensor.front == 0x80))
        handle->left();
    else if ((sensor.front == 0x03))
        handle->right();
    else
        handle->center();
    accel->on();
}
    
```

코스 인식 알고리즘에 의해 인식된 코스에 대해 적절한 동작을 수행한다. TMO의 경우 일반적인 절차적 프로그래밍 언어와 달리 실시간 객체가 병행적으로 동작함으로써 절차적 동기화가 필요하다. 각 코스의 단계별 스텝에 의해 무인 자동차가 동작할 수 있도록 한다.

4.1 코스 주행 알고리즘

5. 결론

TMO에 의한 실시간 제어 시스템의 모델링은 이미 원자력 발전소 냉각수 주입 계통에 대한 시뮬레이션 시스템 구현으로 그 효율성이 입증된 바 있다.

본 논문에서는 실시간 제어 시스템에 적합한 모델로 개발된 마이크로 임베디드 운영체제인 TMO-eCos를 기반으로 무인 자동차 제어 프레임워크와 이를 활용한 응용 모델에 대해 기술하였다. TMO 모델을 이용한 프레임워크는 실시간 제어 시스템 소프트웨어의 규격적 컴포넌트 기반 개발을 제공하고, 나아가서는 비주요 모듈 빌더를 활용한 프로그램 생성의 기반을 제공하므로, 지능형 로봇 소프트웨어의 기본 플랫폼으로 적합한 엔진이라 할 수 있다. 이러한 TMO-eCos는, 기 개발된 TMO-Linux와 함께 대상 플랫폼 H/W를 확장하여, 이 기종의 많은 임베디드 장치들을 연결하는 제어 네트워크 구성에 활용될 수 있을 것이다.

참고문헌

- 1] Kim, J. G., et al, "TMO-Linux: A Linux-based Real-time Operating System Supporting Execution of TMO's," Proc. IEEE Int'l Symposium, ISORC2002, Washington DC, Apr. 28, 2002.
- 2] Jo, E. H., Kim, M. H., Kim, J. G., "Framework for Development of Multimedia Applications Based on the TMO Structuring Scheme," Proc. IEEE Int'l Symposium, ISORC2003. Hakodate, Hokkaido, Japan, May 14-16, 2003.
- 3] Kim, J. G., Kim, M. H., et al, "A Concurrent Event-Driven Simulation Technique for the ROK Army's War Game Model using TMOs," Proc. The 7th World Conference on Integrated Design & Process Technology, Austin, Texas, Dec. 3, 2003.
- 4] Kim, J. G., Kim, M. H., et al "An eCos-based Real-time Micro Operating System Supporting Execution of TMO's" 8th IEEE International symposium, ISORC2005, Seattle May 18-20, 2005