

# 실시간 객체 TMO를 이용한 로봇 제어 모델링

곽 상훈\*, 김현주 서한석 김정국

한국외국어대학교 컴퓨터 및 정보통신공학부

{ csekwak\*, kimhj, hanseok , jgkim}@hufs.ac.kr

## Modeling of a Robot Control System using the Real-time Object Model : TMO

Sang-Hoon Kwak\*, Hyun-joo Kim, Han-seok Seo, Jung-Guk Kim

Department of Computer Science and Engineering, Hankuk University of Foreign Studies

### 요 약

본 논문에서는 분산 실시간 객체 엔진을 내장하는 TMO-LINUX/ARM 커널 상에서, 각종 센서 및 모터 제어로 구성되는 로봇 제어 시스템을 동적 분산 실시간 객체인 TMO로 모델링하고 구현하였다. TMO(Time-triggered Message-triggered Object)는 시간적 조건에 의해 구동되는 객체 내의 실시간 스레드와 메시지에 의해 구동되어 주어진 데드라인 안에 작업을 수행하는 메시지 구동 실시간 스레드들의 동적 멤버들로 구성되는 실시간 객체로, 외부의 입출력에 반응하며 내부 장치에 대한 실시간 작업을 병행해야 하는 로봇 제어 시스템에 대해 매우 적합한 모델이다. 본 논문에서는 이러한 로봇 제어 시스템 설계에의 TMO의 적용이 잘 정의되고 규격적인 설계 모델을 제공함을 보인다.

### 1. 서론

실시간 제어를 요구하는 많은 임베디드 응용에 대한 개발 도구 또는 모델은 소형 임베디드 운영체제의 커널 API가 제공하는 스레드 모델과 각종 타이머에 의한 동기화 도구가 대부분이다. 이러한 하위 수준 모델을 이용한 실시간 제어 시스템은 그 모델링의 자유도가 높아, 규격적 타이밍 제어의 결여, 병행성에 의한 오류, 객체와 스레드의 분리에 의한 유지 보수성의 어려움 등을 갖게 된다. TMO(Time-triggered Message-triggered Object) 모델은 규격적 실시간 프로그래밍 설계를 위해 제안된 모델로 객체 지향 프로그래밍, 시간과 메시지에 의해 구동하는 동적 실시간 스레드의 객체 멤버화, 데드라인 스케줄링 및 분산 IPC 등을 통합적으로 제공하는 모델로 실시간 제어에 대한 분석 가능한 규격적 모델을 제공한다. 본 논문에서는 TMO의 실행 엔진을 리눅스 커널 내에 내장한 TMO-Linux 상에서, TMO 모델을 이용하여 로봇 제어 시스템을 설계 구현한 것으로 로봇 제어 시스템과 같은 시간적 조건과 비동기적 사건을 다루어야 하는 실시간 제어 시스템에 TMO에 의한 모델링이 효율적임을 보여 준다.

본 논문의 2 장에서는 TMO-LINUX<sup>1</sup>에 대해 기술하며, 3 장에서는 TMO-LINUX/ARM 기반의 로봇 제어 응용의 TMO에 의한 모델링 기법을 기술하고, 4 장에서 결론을 맺는다.

### 2. 관련연구

#### 2.1. TMO 모델

분산 실시간 객체 모델 TMO(Time-triggered Message-triggered Object)[1,2,3]는 실시간 시스템의 시간 보장성 컴퓨팅 설계를 지원하기 위한 목적으로 제안되었다. TMO의 구조는 그림 1에 표시된 바와 같이 일반 객체와는 그 구조에 있어서 몇 가지 다른 점을 가진다. TMO는 크게 ODS(Object Data Store), 시간 구동 메소드 (Time-triggered Method: Spontaneous Method: SpM) 그룹, 그리고 메시지 구동 메소드(Message-triggered Method: Service Method: SvM) 그룹의 세 요소들로 구성된다. 다음은 TMO의 특징을 요약한 것이다.

- TMO를 구성하는 한 형태의 멤버인 시간 구동

<sup>1</sup> 2001년 한국외국어대학교 RTDCS Lab.에서 개발

메소드(SpM)는 객체 내의 동적 메소드로, 객체의 멤버의 특성을 갖는 스레드로 구현되어 주어진 시간 조건에 의해 자율적으로 구동 된다. SpM에는 시작과 종료 시간, 주기, 각 주기 구동시의 데드라인으로 구성되는 시간 조건(AAC: Autonomous Activation Condition)이 주어지고, 커널이나 TMO 엔진은 이에 대한 데드라인 기반 스케줄링을 제공하여야 한다.

- TMO를 구성하는 또 다른 한 형태의 멤버인 메시지 구동 메소드(SvM)는 원격 또는 국부 TMO의 메소드로부터의 IPC 메시지 수신에 의해 구동되며, 메시지에 대한 서비스 완료 시까지 데드라인 스케줄링이 적용된다. SvM 역시 객체의 멤버의 특성을 갖는 스레드로 구현되며, 커널이나 TMO 엔진은 이에 대한 데드라인 스케줄링을 제공하여야 한다. SvM의 데드라인은 스케줄링에 반영되는 의미와 함께, SvM의 클라이언트가 데드라인 보다 더 작은 빈도로 메시지를 보내면 시간 내의 서비스가 보장되지 않음을 의미한다.
- SvM을 구동시키는 IPC 메시지는 분산 환경에도 변화 없이 그대로 적용되는, 네트워크에 투명한 분산 IPC이다. 따라서 TMO 들의 네트워크로 표현되는 실시간 시스템은 그 구성의 변화 없이 그대로 단일 노드 컴퓨팅이나 분산 컴퓨팅에 적용된다.

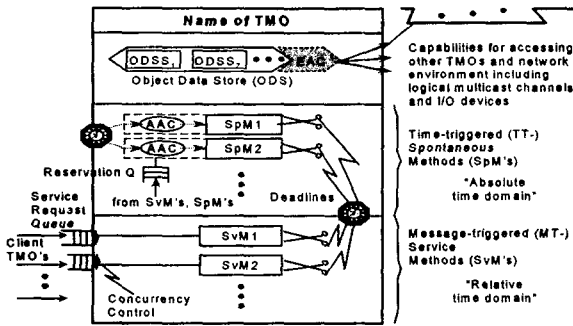


그림 1 TMO의 구조

2.2. LINUX 상의 로봇 제어 기반 구조

기존의 LINUX 커널 기반의 로봇제어 구조[4]는 제어를 위한 응용 프로그램 및 장치 제어 라이브러리 부분과 Linux 커널 내부의 USB기반의 각종 장치 드라이버들로 구성된다. 제어 응용은 모션/센서 라이브러리를 사용하여 제어 응용을 작성한다. 제어 라이브러리 함수들의 동작형태는 Linux Kernel의 USB용 모션/센서 디바이스 드라이버를 통하여 전달된 명령을 해석하고 각각의 축에 연결된 모터와 각각의 센서 등을 구동하는 방식을 사용한다. 제어 응용은 입력 장치들인 소나 센서 등의 상태를 폴링을 통해 입력 받는다.

3. TMO를 이용한 로봇 제어 시스템의 모델링

3.1. TMO 기반 로봇 제어 응용의 설계

TMO 모델을 이용한 제어 응용은 주기적으로 실행되는 TMO 객체 내의 실시간 스레드 SpM과 산발적 이벤트를 처리하는 SvM으로 구현되고, TMO의 ODS는 이들 스레드들이 공유하는 로봇 제어 시스템의 실시간 자료를 유지한다. SpM들은 센서 라이브러리를 통해 각종 센서의 상황을 주기적으로 폴링하는데 사용되며 SvM은 ODS의 상황이나 SpM의 요구에 의해 로봇을 구동하는데 사용된다.

TMO 기반의 로봇 제어 응용의 모델은 그림 2와 같다.

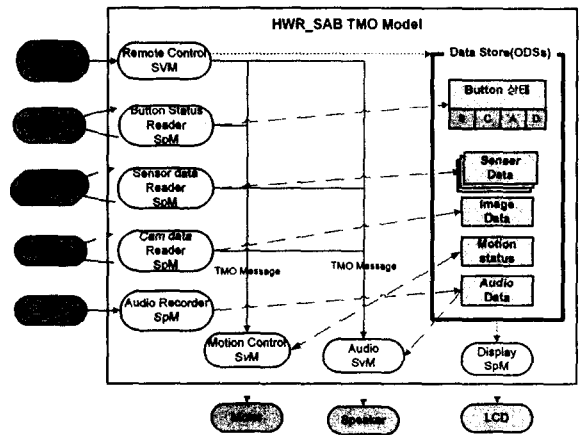


그림 2 TMO 기반의 로봇제어 응용 모델

로봇 제어 응용을 구성하는 TMO 내의 메소드들은 다음과 같다.

- Remote-Controller SvM  
외부 네트워크로부터 로봇 제어에 대한 명령을 수신하여 ODS에 기록하고 Motion Control 및 Audio Control을 위한 TMO Message를 담당 SvM에 전송한다.
- Button-Status-Reader SpM  
로봇 시스템의 입력 장치인 5 개의 버튼 입력을 주기적으로 확인하고 Button 상태를 ODS에 기록한다.
- Sensor-Data-Reader SpM  
그림 2의 Button 상태의 Sensor bit(S)이 1로 셋팅되어 있는 경우 Sensor로부터 주위 물체의 거리를 측정하여 ODS에 저장하고, 로봇의 동작이 필요한 경우 Motion-Controller SvM에 요구 메시지를 전송한다.
- Cam-Data-Reader SpM  
그림 2의 Button 상태의 Cam bit(C)를 확인하여 1로

셋팅되어 있으면, Cam 이미지를 읽어 ODS 영역에 이미지를 저장한다. 저장된 데이터를 분석한 후, 로봇 동작이 필요한 경우, Motion-Controller SvM에 요구 메시지를 전송한다.

- Audio-Recorder SpM  
그림 2의 Button 상태의 Audio bit(A)를 확인하여 1로 셋팅 되어 있으면, 임의 시간 동안 ODS에 audio 자료를 저장하고 Audio-SvM에 메시지를 전송한다.
- Motion-Controller SvM  
Remote-Controller, Button-Status-Reader, Sensor-Data-Reader, Cam-Data-Reader로부터 메시지를 받아 Motion 상태를 확인한 후 Motor를 구동한다.
- Audio SvM  
Audio-Recorder, Remote-Controller로부터 메시지를 받아 ODS에 있는 audio 자료를 스피커를 통해 출력한다.
- Display SpM  
그림 2의 Button 상태의 Display bit(D)를 확인하여 1로 셋팅되어 있는 경우 ODS의 정보를 주기적으로 LCD 화면에 갱신한다.

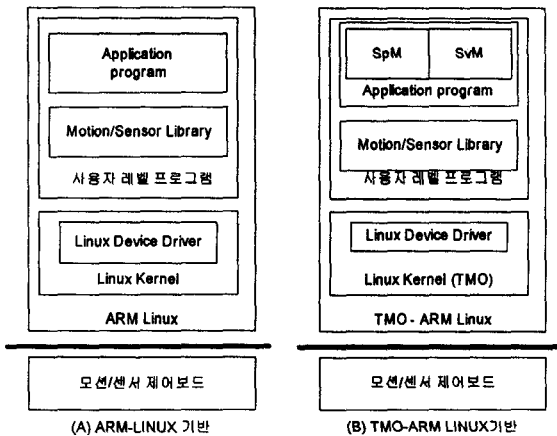


그림 3 기존 제어 구조와 TMO-Linux 플랫폼을 사용한 제어 구조

### 3.2. 구현 및 실험

TMO를 이용하여 설계된 로봇 제어 시스템을 구현 실험하기 위해, 기존의 x-86 기반으로 개발된 TMO-LINUX는 ARM CPU를 사용하는 로봇 제어보드 플랫폼에 포팅되었다(TMO-Linux/ARM, 그림 3).

포팅된 TMO-Linux/ARM 커널은

- TMO 및 SvM, SpM 들의 실행 지원
- 데드라인 스케줄링
- 실시간 스레드의 ODS 공유 접근 제어를 위한 LOCK 기능

들을 제공한다.

구현된 시스템은 한울 로보틱스의 HW-RD에 탑재되어 현재 구동 중이며, TMO 객체와 두 가지 성격의 내부 매소드들로 구현되어 각종 디바이스의 탈 부착 시, 모듈성을 바탕으로 한 유연한 유지보수성을 보이고 있다.

### 4. 결론 및 향후과제

본 논문에서는 TMO의 실행 엔진을 리눅스 커널 내에 내장한 TMO-Linux 상에서, TMO 모델을 이용하여 로봇 제어 시스템을 설계 구현한 것으로, 로봇 제어 시스템과 같은 시간적 조건과 비동기적 사건을 다루어야 하는 실시간 제어 시스템에 TMO 객체 내의 두 가지 실시간 스레드 모델인 SpM과 SvM이 모델링에 효율적임을 보여 준다. SpM은 각종 센서에 대한 주기적 실시간 폴링에 의한 상태 입력에 사용되는 실시간 스레드로, SvM은 외부 이벤트나 내부 이벤트 발생 시의 실시간 처리를 담당하는 스레드로 활용하여, 로봇 제어와 같은 실시간 제어 시스템에 규격적인 모델링이 가능함을 보였다. 이와 같은 TMO에 의한 실시간 제어 시스템의 모델링은 이미 원자력 발전소 냉각수 주입 계통에 대한 실시간 시뮬레이션 시스템 구현으로 그 효율성이 입증된 바 있다.

향후 연구로 필요한 부분은 현재 TMO-Linux/ARM 상에서 구현된 로봇 제어 시스템의 TMO 지원 커널을 슬림화하여 100KB 정도의 TMO 지원 마이크로 커널을 개발 탑재하는 것이다. 이러한 연구는 현재 ITRC 과제의 일환으로 진행되고 있다.

### 참고 문헌

- [1] Kim, K.H. and Kopetz, H., "A Real-Time Object Model RTO.k and an Experimental Investigation of Its Potentials", Proc. 18th IEEE Computer Software & Applications Conference, pp.392-402, November 1994.
- [2] Kim, K., "Object Structures for Real-Time Systems and Simulators", IEEE Computer, August 1997, pp.62-70.
- [3] Kim, K.H., "Group Communication in Real-Time Computing Systems: Issues and Directions", Proc. FTDCS '99 (7th IEEE Workshop on Future Trends of Distributed Computing Systems), Cape Town, South Africa, Dec. 1999, pp.252-258.
- [4] HANOOL ROBOTICE "HWR - RD ver.1.0, pp.54.