

AUTOSAR 기반 가상 ECU를 통한 태스크 동작 시각화

조수호¹, 조정훈²

¹경북대학교 전자공학부 석사과정

²경북대학교 전자공학부 교수

sooho0911@gmail.com, jcho@knu.ac.kr

Task Execution Visualization Using AUTOSAR-Based Virtual ECU

Su-Ho Jo¹, Jeong-Hun Cho²

¹Dept. of Electronic Engineering, Kyungpook National University

²Dept. of Electronic Engineering, Kyungpook National University

요약

최근 AUTOSAR 기반 차량용 소프트웨어 개발에서 태스크 (Task)의 동작 흐름과 상태 전이를 실시간으로 파악하는 것은 시스템의 안정성과 타이밍 보호 (Timing Protection)를 보장하는 데 매우 중요하다. 특히 가상 ECU (Virtual ECU) 환경에서는 실제 하드웨어 없이도 에뮬레이션 기반의 분석이 가능하다는 장점이 있어, 개발 초기 단계에서의 오류 검출 및 최적화에 유용하다. 본 논문에서는 vECU 에뮬레이터를 활용하여 AUTOSAR OS 태스크의 실행 로그를 수집하고, 이를 시각적으로 분석하기 위한 실시간 Visualizer 툴을 설계 및 구현하였다.

1. 서론

최근 차량용 소프트웨어는 고도화 및 대형화가 진행됨에 따라 시스템 내부의 복잡성이 급격히 증가하고 있다. 특히 AUTOSAR 기반 전자제어장치 (ECU)에서는 실시간 운영체제 (RTOS) 하에 다양한 태스크 (Task)가 주기적 실행, 이벤트 트리거, 우선순위 및 선점 관계에 따라 복잡하게 얹혀 동작한다.

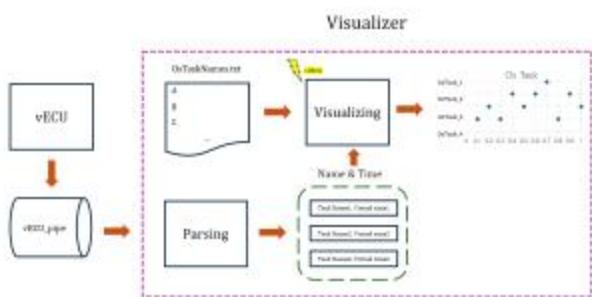
AUTOSAR 기반의 시스템에서 태스크의 타이밍 보호와 정확한 스케줄링 설계는 실시간성 보장을 위한 핵심 요소이다 [1]. 특히 하드 실시간 태스크는 주기, 지터, 테드라인 등의 타이밍 제약을 만족하지 못할 경우 기능 오류나 안전 문제로 이어질 수 있다.

하지만 실제 ECU는 제한된 디버깅 기능과 물리적 제약으로 인해 태스크 실행 정보를 외부에서 효과적으로 관찰하거나 분석하기 어렵다. Hardware-In-the-Loop (HIL) 시뮬레이션이나 실차 기반 테스트는 높은 비용과 많은 시간을 요구하며, 반복적이고 정밀한 디버깅에는 한계가 존재한다.

본 연구에서는 이러한 문제를 해결하기 위해, 가상 ECU (vECU) 환경에서 AUTOSAR의 태스크 상태를 실시간으로 수집하고 이를 시각적으로 표현하는 시스템을 제안한다 [2]. 제안된 시스템은 실제 ECU와 동일한 소프트웨어 스택을 기반으로 동작하여 현 실성과 신뢰성을 확보하며, 태스크 간 선점 관계, 상태 전이, 우선순위 변동 등을 그래픽 형태로 직관적으로 분석할 수 있도록 지원한다. 이를 통해 소프트웨어 개발 초기 단계부터 동시성 오류와 타이밍 이슈를 조기에 발견하고 해결할 수 있는 기반을 제공한다. 본 논문에서는 제안 시스템의 설계 및 구현 방법을 소개하고, 사례 기반 실험을 통해 그 유효성과 실용성을 검증한다.

2. 가상 ECU를 통한 태스크 동작 시각화

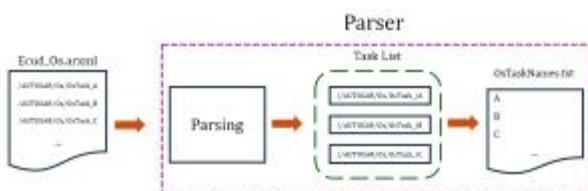
가상 ECU 환경은 vECU를 기반으로 구성되었으며, AUTOSAR CP (classic platform)인 Mobilgene을 기반으로 S32K148 보드의 base project를 사용하였다 [3]. 그림 1은 본 연구에서 진행될 전반적인 시스템 구성도이다.



(그림 1) 시스템 전체 구성도

시각화를 위해서는 실행 중인 태스크 이름을 사전에 파악해야 한다. 이를 위해 AUTOSAR의 arxml 파일을 기반으로 태스크 정보를 추출하는 파서가 필요하였다. arxml 파일에서 태스크 정보에 대한 구조를 파악한 후, 그림 2와 같이 파서(parser)를 설계하였다. 주요 동작은 다음과 같다.

- (1) 경로를 따라 arxml 내의 태스크 객체를 탐색한다.
- (2) 태스크 이름을 추출하여 리스트화 한다.
- (3) 결과를 텍스트 파일로 저장하여 후속 시각화 단계에서 y축 레이블로 활용 한다.



(그림 2) 파서 구성도

이 파서는 Python 기반으로 구현되었으며, 향후 다양한 AUTOSAR 프로젝트에서 재활용 가능하다.

이후 실시간으로 로그를 수집하기 위해 본 시스템은 네임드 파일(named pipe)을 활용한 통신 구조를 채택하였다.

네임드 파일은 유닉스 계열 운영체제에서 지속적인 FIFO (First-In-First-Out) 통신 채널을 의미하며, 프로세스 간에 실시간 데이터 스트림 전달이 가능하다는 장점이 있다.

vECU 에뮬레이터는 태스크 실행 로그를 vECU_pipe라는 파일 기반 파일을 통해 출력하고, 시각화 도구는 해당 파일을 주기적으로 모니터링 하며 새로운 로그가 기록될 때마다 이를 파싱하여 시각화에 반영한다.

이를 통해 파일 I/O 지연 없이 실시간 수준의 로그

반영이 가능하며, 각 태스크의 실행 시점과 흐름을 빠짐없이 추적할 수 있다.

Visualizer는 실시간 분석을 위해 PyQt 기반의 시각화 도구를 설계하였다. 주요 기능은 다음과 같다.

- 첫째, vECU에서 출력한 로그를 앞서 생성한 파일(vECU_pipe)을 통해 읽어 들인다.
- 둘째, parse_task_and_time() 함수를 사용하여 로그에서 태스크 이름과 실행 시점을 추출한 후, 이를 리스트 형태로 저장한다.

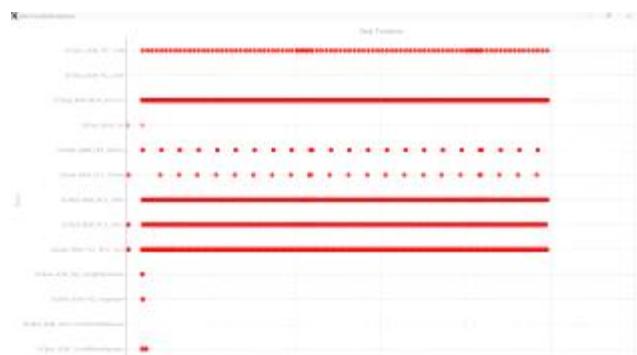
셋째, 기존에 생성한 태스크 이름 리스트(텍스트 파일)를 기반으로 y축 레이블을 설정하며, x축은 실행 시간을 나타내며 초(sec) 단위로 설정한다.

넷째, 타이머를 활용하여 100ms 간격으로 로그를 체크하고, 새로운 데이터가 있을 경우 자동으로 업데이트하여 실행 상황을 반영한다.

이를 통해 사용자는 가상 ECU 환경에서 동작하는 각 태스크의 실행 타이밍을 실시간으로 확인할 수 있다.

여러 개의 AUTOSAR 태스크가 실행되는 시나리오를 구성하여, 해당 태스크들이 vECU 로그에 남기는 호출 타이밍을 시각화하였다. 그림 3은 정상적인 태스크 호출 예제, 그림 4는 비정상적인 태스크 호출 예제에 대한 실행 결과이다.

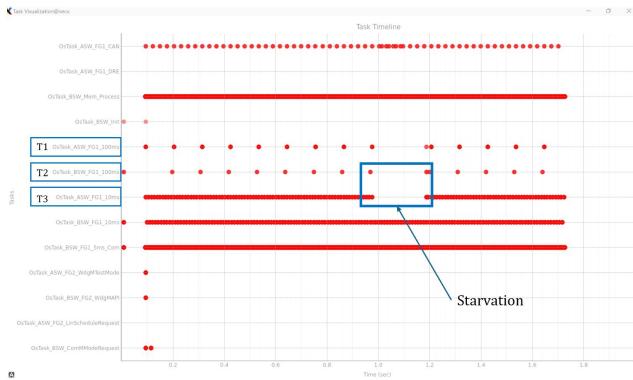
이 예제에서는 주기적으로 동작하는 태스크 중 OsTask_BSW_FG1_100ms (T2)와 OsTask_ASW_FG1_10ms (T3)만 OsTask_ASW_FG1_100ms (T1) 보다 낮은 우선순위를 가지도록 설정하였다.



(그림 3) 정상적인 태스크 실행 결과

그림 3의 그래프는 시스템이 정상적으로 동작하는 경우로, 각 주기 태스크가 설정된 주기에 따라 안정

적으로 반복 실행되며, 낮은 우선순위를 가진 태스크들도 간섭 없이 주기적으로 실행됨을 확인할 수 있다.



(그림 4) 비정상적인 태스크 실행 결과

반면, 그림 3의 그래프는 우선순위가 높은 주기 태스크의 실행 시간이 설정된 주기보다 길어지는 경우를 보여준다.

이 경우, 스케줄러는 높은 우선순위를 가진 태스크 T1 이 지속적으로 선점하게 되며, 그 결과 우선순위가 낮은 태스크인 T2 와 T3 는 실행 기회를 얻지 못하고 지속적으로 대기 상태에 머무르게 된다.

이는 AUTOSAR OS의 우선순위 기반 비선점형 스케줄링 정책에 따라 자연스럽게 발생하는 현상으로, 태스크 실행 시간 증가가 다른 태스크의 starvation (기아 상태)으로 이어질 수 있음을 시각적으로 입증한 사례이다.

이러한 시각적 분석은, 실제 시스템 설계 시 태스크 실행 시간 분석과 우선순위 설정의 중요성을 강조하며, 타이밍 보호 (Timing Protection)나 스케줄링 개선에 대한 기초 자료로 활용될 수 있다.

3. 결론 및 추후 연구

본 논문에서는 AUTOSAR 기반의 가상 ECU 환경에서 동작하는 태스크의 실행 흐름을 실시간으로 확인하기 위한 시각화 도구를 개발하였다. vECU 에뮬레이터와 파이썬 기반의 로그 파싱/시각화 기술을 연계하여, 시스템의 동작을 직관적으로 분석할 수 있는 기반을 마련하였다. 향후에는 태스크 우선순위, 상태 전이 (Task State Transition) 분석, 타이밍 보호 위반 감지 기능 등을 추가함으로써, 실시간 시스템 분석 도구로서의 활용 가능성을 더욱 높일 수 있을 것으로 기대된다.

이 논문은 2025년도 정부(산업통상자원부)의 재원으로 한국산업기술진흥원의 지원을 받아 수행된 연구임
(RS-2024-00415938, 2024년
산업 혁신 인재 성장 지원 사업)

참고문헌

- [1] S. Anssi, S. Tucci-Piergiovanni, S. Kuntz, S. Gérard, and F. Terrier, “Enabling Scheduling Analysis for AUTOSAR Systems,” in *Proc. 14th IEEE ISORC*, 2011, pp. 152 - 159.
- [2] M. Stolpe, “AUTOSAR Application Visualization,” BSc Thesis, Dept. of Computer Science and Engineering, Chalmers Univ. of Technology, Gothenburg, Sweden, 2014.
- [3] H. Kim, J. Kwak, and J. Cho, “AUTOSAR-Compatible Level-4 Virtual ECU for the Verification of the Target Binary for Cloud-Native Development,” Electronics, vol. 13, no. 18, p. 3704, 2024. [Online]. Available: <https://www.mdpi.com/2079-9292/13/18/3704>.