

과목명	자동차 임베디드 시스템 2	담당교수	서석현 교수님
학생이름	김세환	학번	2015146007
(작성유의사항)			
1. 다음 과제에서 요구하는 사항을 정리하여 작성하세요.			
2. 수업내용 중 질문사항을 정리하여 작성하세요.			
과제	1. 차량 플랫폼과 전기전자아키텍처의 도입 배경과 특징을 설명하시오.		
	<p>차량은 그 어느때보다 발전하고 복잡해졌다. 차량을 컨트롤하는 ECU 또한 발전하고, 하나의 차량에 들어가는 ECU의 개수 및 소프트웨어의 복잡성도 증가하고 있다.</p> <p>차량 플랫폼의 도입은 이러한 복잡한 차량을 설계함에 있어서 차량의 기본 성능을 결정하는 주요 시스템들을 플랫폼화 하는 것을 의미한다. 이로써 새로운 차량을 개발할 때 부품 공용 극대화에 의한 원가 절감, 규모의 경제 달성 및 생산 라인의 개/보수가 불필요해져 생산에 대한 유연성이 확보된다.</p> <p>또한 플랫폼을 상세 설계하기 이전에 충분한 개발과 검증을 통해 해당 플랫폼을 공유하는 차량들은 충분히 검증되고 안정된 상태로 출시가 가능해진다.</p> <p>무엇보다도 플랫폼 교체를 계획적이고 충분히 실험적으로 진행 할 수 있는 점이 큰 도입배경인 것 같다.</p> <p>관련 업계 종사자에게 이야기를 들은 내용중 다음이 인상깊었다.</p> <p>새로운 차량 플랫폼은 이미 차량 제조사들이 10~15년치를 개발해두었고, 그에따른 금속 금형등과 같은 생산라인에 필요한 요소들까지도 미리 갖고 있다. 따라서 그들은 충분한 실험과 검증을 통해 시간에 쫓기지 않고 미리 개발하고 검증한 플랫폼을 시장에 선보인다고 한다.</p> <p>차량 전기 전자 아키텍처도입은 복잡해진 ECU간의 통신 및 와이어링, 그리고 ECU내 소프트웨어 등 여러 요소들을 통합하고 최적화 하기 위해서 도입되었다.</p> <p>산업분야는 표준화가 잘 이루어지지 않고 각 메이커마다 독자적인 프로토콜이나(대표적으로 PLC나 제어기들의 Field Bus Protocol들..)하드웨어를 통해 독자 라인을 가져가려고 하는데 이렇게 될 경우 개발하는 입장에서는 정말 난감하다.</p> <p>차량 산업의 경우 차량 전기 전자 아키텍처를 도입하여 전기 전자 시스템을 구조화하고 최적화 하는 전략을 취해 시스템 표준화 및 공용화를 달성하였다.</p> <p>따라서 규모의 경제 달성 및 품질 신뢰성을 확보 가능하였고, 제어기 통합 및 최적 배치를 통한 원가 절감, 배선 회로 축소등 부수적인 효과들도 달성하였다.</p>		

2. 차량용 소프트웨어 플랫폼인 AUTOSAR의 도입 배경과 특징을 설명하시오.

차량용 소프트웨어는 신뢰성, 안정성, 리얼타임 보장, 메모리 등 자원 소모 최소화, 전기적 안정성, 페루프 제어등의 요구 조건을 충족시켜야 한다.

일종의 산업 표준을 제정하고 각 제조사별로 분리된 리소스가 아닌, 공용하여 개발 할 수 있는 환경을 만들고자 2003년 AUTOSAR 개발 파트너쉽이 BMW, 보쉬, 지멘스, 폭스바겐 등에 의해 설립되었다.

표준 자동차 소프트웨어 구조를 설계하고 이를 일련의 `사양서`로 제공하여 여러 차량 제조사들이 사양서에 있는 표준을 따라가도록 제시하기 위해 AUTOSAR 오픈된 플랫폼이 자동차 산업계에 도입되었다.

AUTOSAR 플랫폼은 다음과 같은 특징이 있다.

1. 추후 미래 차량의 안정성, 신뢰성, S/W 업그레이드, 업데이트 및 유지보수성을 만족하도록 설계되었다.
2. 확장성과 유연성이 매우 좋다.
3. 소프트웨어의 모듈화가 가능하도록 설계되었다.
4. 모듈들을 재사용 가능하도록 설계되었다.
5. S/W레이어와 하드웨어간 공용화된 인터페이스를 제공, 개발자는 OS API를 적절히 활용하여 개발에만 신경쓸 수 있다.

3. Full-Preemptive Scheduling과 Non-Preemptive Scheduling 정책의 차이점을 설명하시오.

Full-Preemptive 스케줄링 정책은 어느 Task가 프로세서에서 실행중일 때, 자신보다 우선순위가 높은 Task가 Ready 상태가 되었을 때, 실행중이던 Task를 Ready상태로 전이시키고, 우선순위가 높은 Task를 running으로 상태를 전이시킨다. 이후 우선순위가 높은 running인 Task가 Terminate되었을 때, 기존에 실행중이었던 현재 Ready 상태인 Task를 다시 running상태로 전이시킨다.

즉 현재 실행중인 Task보다 우선순위가 높은 Task가 Ready 상태일 경우 지체 없이 Context swtich를 하여 해당 Task를 수행하는 정책이다.

Non-Preemptive 스케줄링 정책은, 어느 Task가 실행중일 때, 자신보다 우선순위가 높은 Task가 Ready상태이더라도, 실행중이던 Task가 마무리(Terminate)된 이후에 Ready상태인 Task를 수행하는 스케줄링 정책이다.

	<p>4. Basic Task와 Extended Task의 차이점을 설명하시오.</p> <p>Basic Task와 Extended Task의 차이는 waiting이라는 상태가 있느냐 없느냐의 차이이다. waiting이라는 상태는 해당 상태의 Task가 특정 event가 발생되기를 기다리는 상태이며, 특정 event가 발생할 경우 ready로 상태가 전이되는 상태다.</p>
질문사항	<p>ECC2나 BCC2의 경우 Basic Task당 여러번의 Activation이 가능하다고 하는데, 이 의미는 해당하는 Basic Task의 코드가 ready 상태 (우선순위 큐?)에 들어가면(activate) 해당 큐에 하나의 Basic Task가 여러번 들어갈 수 있고, 여러번 실행 될 수 있음을 의미하는 것인가요?</p> <p>Extended Task는 특정한 event를 기다리는 상태인 waiting상태를 가질 수 있는데, 여러 Task가 동일한 event를 기다릴 수 있나요?</p>