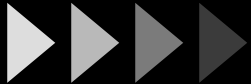


# Control Process



스마트카 협동과정 기석철



충북대학교  
CHUNGBUK NATIONAL UNIVERSITY

E-campus/강의자료  
[sckee@cbnu.ac.kr](mailto:sckee@cbnu.ac.kr)





# Contents

- 1. Overview of Control Process**
- 2. Engineering Process**
- 3. Support & Mgmt. Process**





# Control Process

## ❖ Engineering Process

- 시스템을 구현하기 위한 솔루션을 정의하는 업무와 정의된 솔루션을 구현하는 업무에 대한 절차
- Core Process 라고도 칭함

## ❖ Support & Management Process

- Engineering Process를 원활하게 수행할 수 있도록 관리해야 하는 영역과 지원이 필요한 영역에 관한 업무 절차



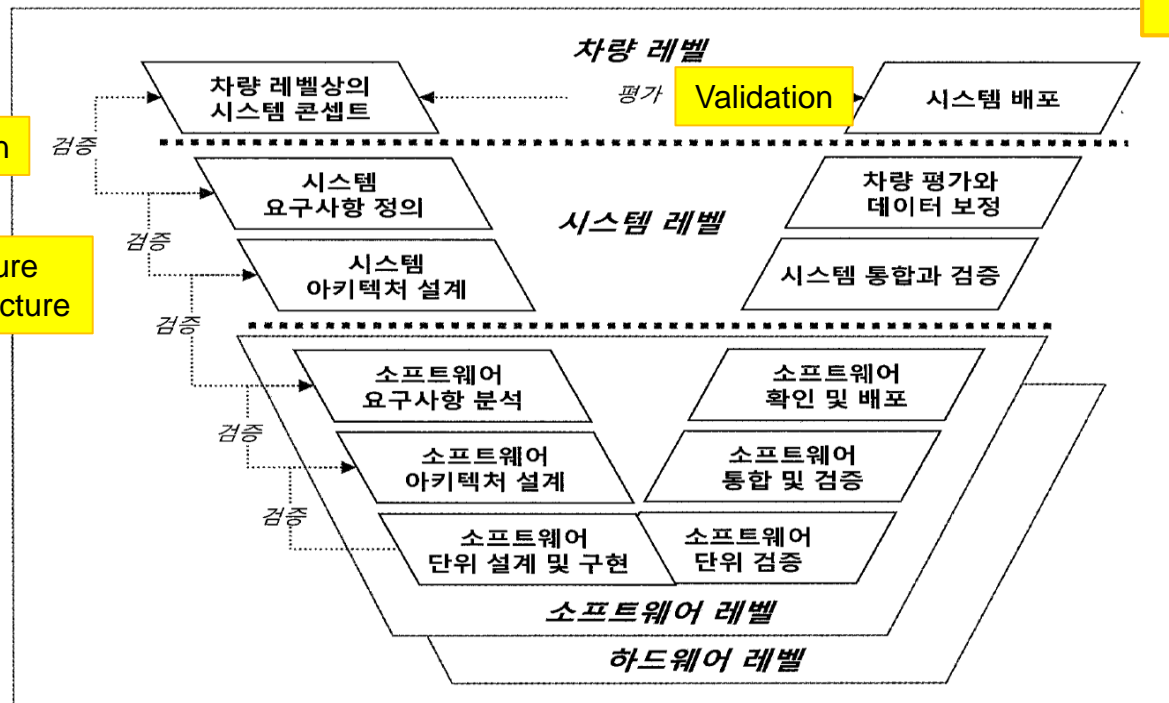


# Engineering Process

## ❖ ECU (Embedded System)

- System, S/W, H/W level로 구분
- 결함이나 변경 요청에 의한 **Feedback**으로 반복
- A → B → C → D Sample 순으로 개발

V - cycle



- Logical Architecture
- Technical Architecture

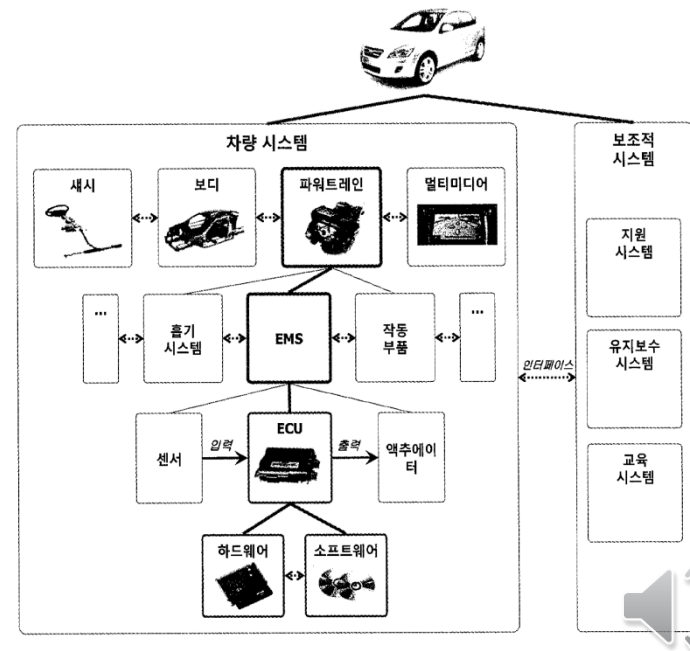
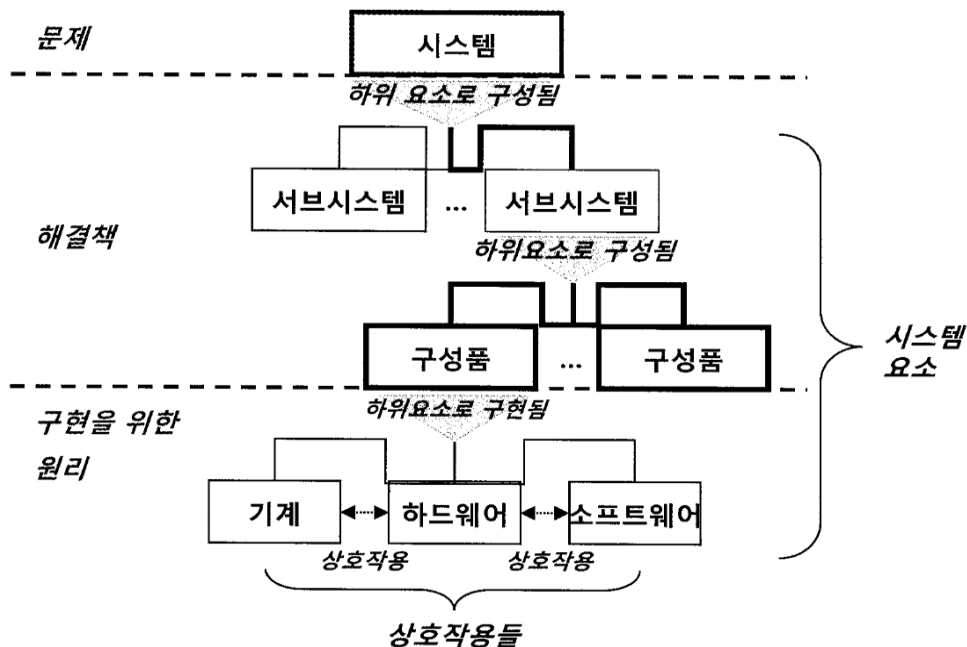


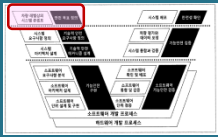


# What is System ?

## ❖ 그리스 어원 ( 'Systema' )

- 조직화된 통일체 ( An organized whole )
- 하나 이상의 목적이 있으며, 목적 달성을 위해 정해진 기능이나 절차에 따라 동작하고 계층구조를 갖는 여러 구성 요소가 유기적인 상호관계를 통해 목적을 달성하는 것

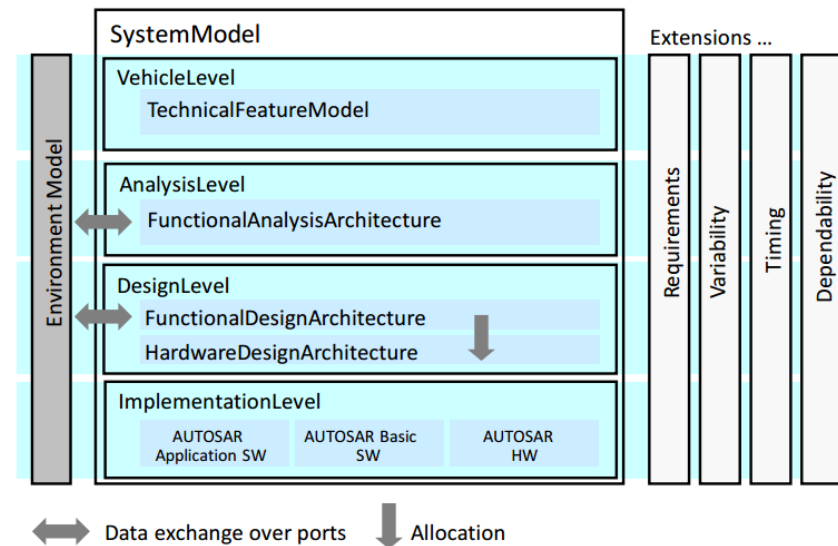




# Vehicle level System Concept



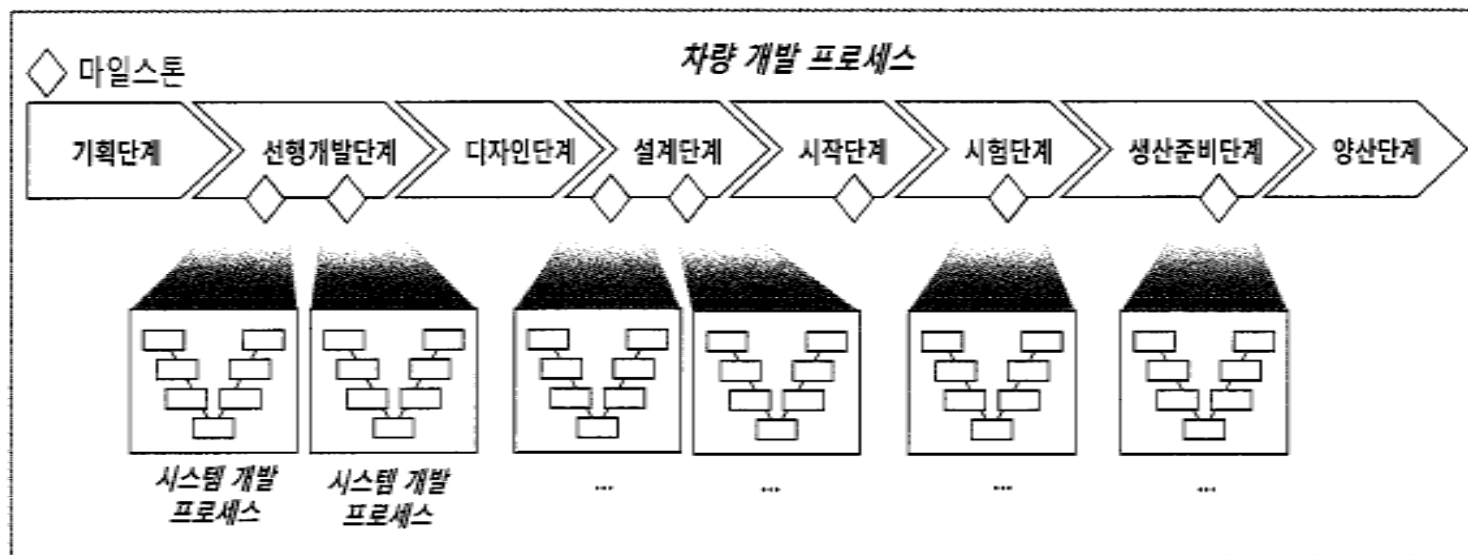
- ❖ 차량 시스템의 기능을 분석해 임무를 할당하는 작업
  - 차량 상태, 운전모드, 부가기능, 연비목표, 안전목표 등의 성능이나 품질
  - **EAST-ADL :**
    - Architecture Description Language (ADL) initially defined in the ITEA project EAST-EEA around 2000.
    - AUTOSAR를 포함하는 상위 레벨 개념
    - EAST-ADL is an approach for describing automotive electronic systems through an information model that captures engineering information in a standardized form





## 차량개발 프로세스와 엔지니어링 프로세스

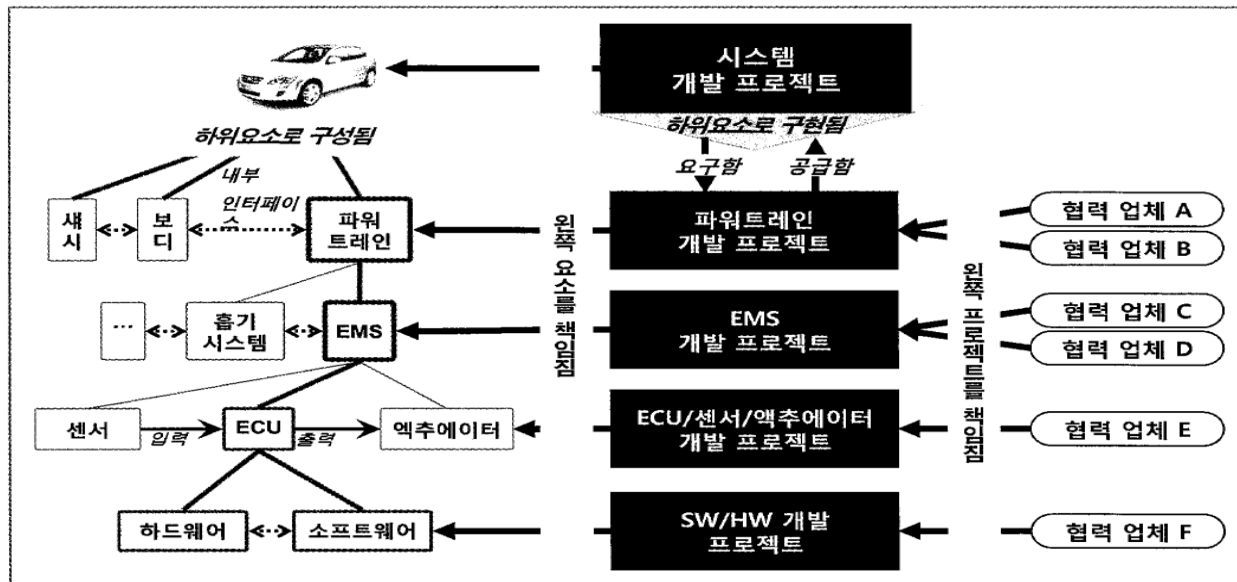
- ❖ 차량 개발 단계별로 엔지니어링 프로세스 반복
- ❖ Trial Car → A sample, Proto Car → B sample, Pilot Car → C sample, Production Car → D sample
- ❖ 개발 보조 활동
  - Version, Tracking, Data Calibration Management, etc



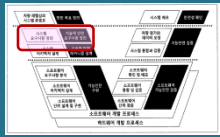


# 협업 관리

- ❖ WBS(Work Breakdown Structure) Define
- ❖ 산출물 (Deliverables) 지원 및 관리 계획 수립
- ❖ Deliverables 배포 기준과 절차 정의 및 평가 계획 수립
- ❖ 착수 시점 대비 변경 발생 시 계획 조정 및 공유 절차도 정의 되어야 함







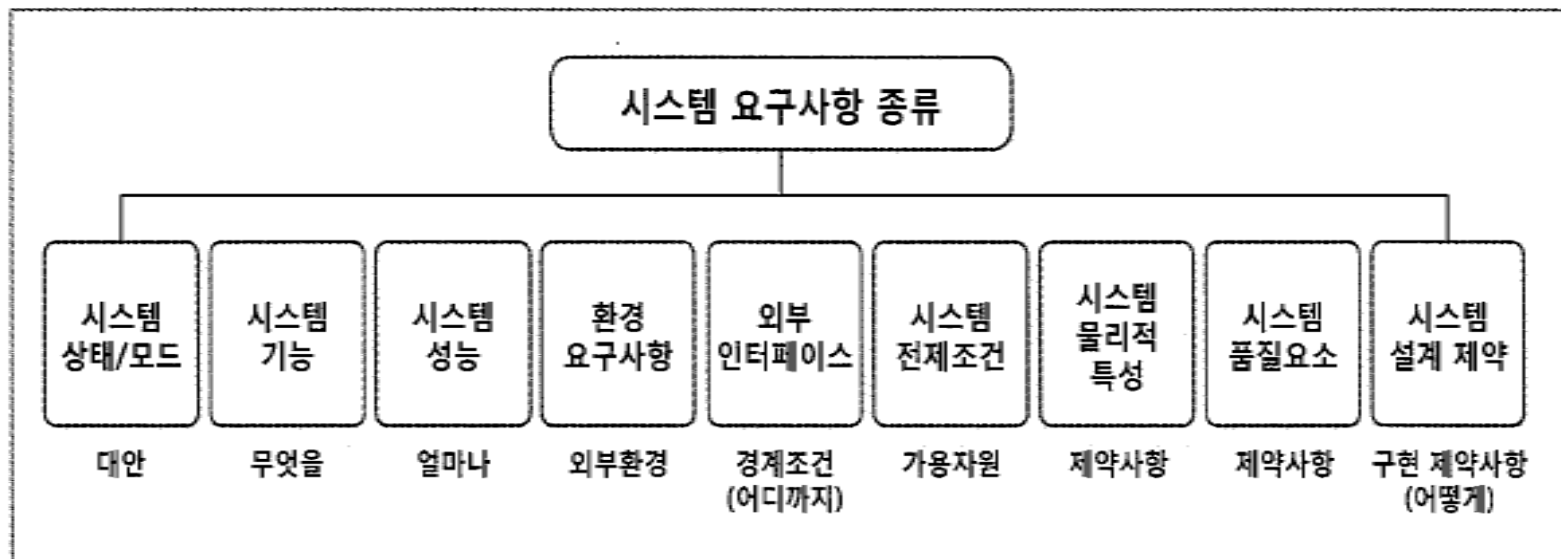
# System Requirement Definition

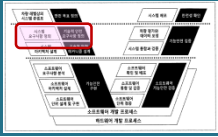


## ❖ 언제, 무엇을 얼마나 수행해야 하는지 시스템 관점에서 정의 필요

- 시스템 운용 관점 : 차량의 상태와 동작 모드에 따라 시스템이 수행해야 할 역할과 기능 분석을 요구사항으로 정의
- 시스템 관점, 기술적 관점

## ❖ 시스템 요구 사항의 종류

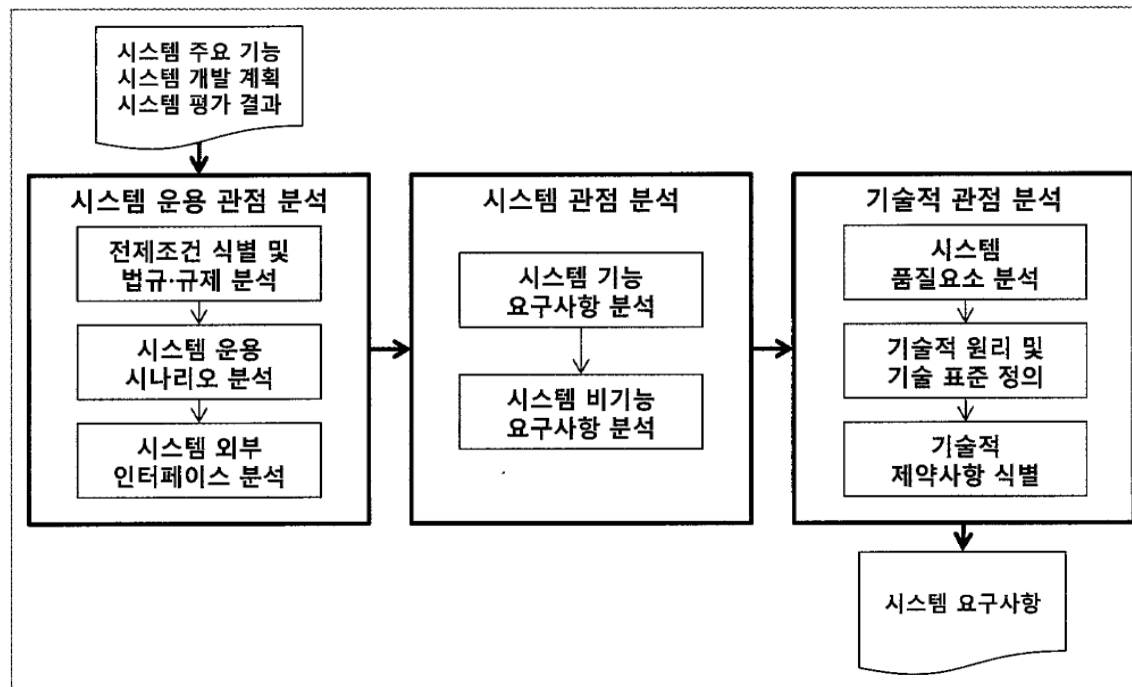




# System Requirement Definition



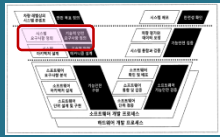
## ❖ Process for system requirement definition



LKAS Camera SR

- Function : Driving, Braking, Steering, etc
- Non-Function : Response time, Memory consumption, Reliability, etc

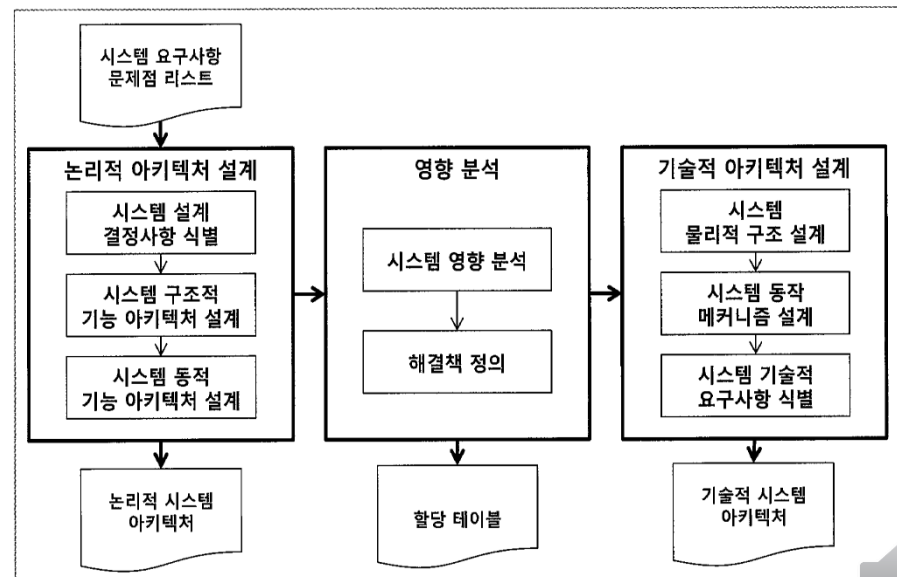
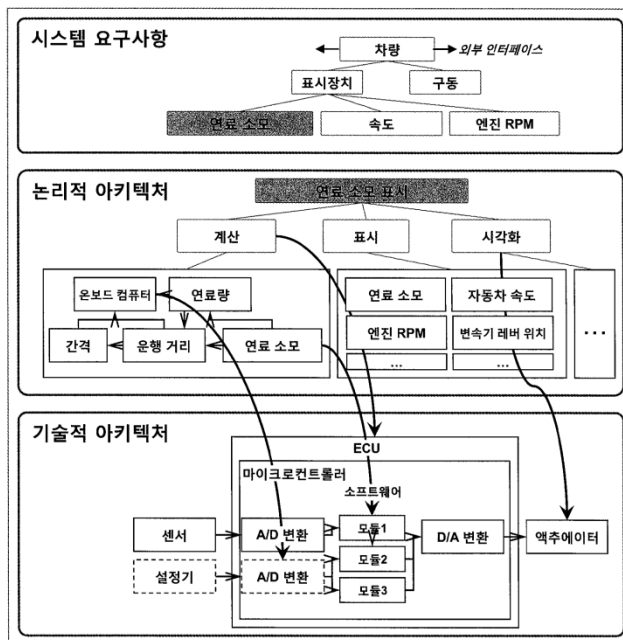


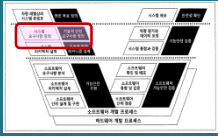


# System Architecture Design



- ❖ 시스템 요구사항과 아키텍처 설계는 시간에 따라 서로 정합성을 맞춰가면서 변경하고 개선하는 개념
- ❖ 논리 아키텍처 설계 후 영향 분석으로 선정된 최상의 해결책을 기반으로 기술적 아키텍처 설계를 수행
- ❖ 기술적 아키텍처가 완성되면 시스템 엔지니어링 사양을 정의하고 하위 구성요소들의 요구 사항을 정의



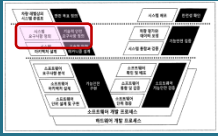


# Logical Architecture Design



- ❖ 논리적으로 수행해야 하는 기능과 기능들간의 관계 정의
  - 복잡한 시스템을 명확하게 정의하고 개발 기간 산정
  - 시스템 사용관점의 요구사항과 제약 사항을 분석해 기능 도출하고 기능별 인터페이스 정의
  - 구조적 기능과 동적 기능 아키텍처로 구분
  - 추상적인 개념의 기능 네트워크(function network) 설계
  - **Deliverable : formal architecture model**



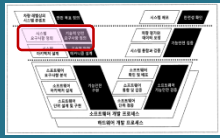


# Effect Analysis



- ❖ 기술적 가능성과 영향 분석을 통해 각각의 기능을 구현할 해결책 정의
  - 시스템의 비기능 요구사항과 식별된 설계 결정 사항을 고려해 영향 분석 수행해 소프트웨어나 하드웨어 등 기술적으로 구현 가능한 최적의 솔루션 정의
  - 설계 결정 사항의 우선순위에 따라 해결책을 선택할 수 있음
  - 재사용 대상 선정 : 과거 유사 시스템 개발 사례 검토
  - 신기술 대상의 경우 신기술 가능성과 관련 기술 정의





# Technical Architecture Design



## ❖ 기술적으로 구현해야 하는 해결책 정의

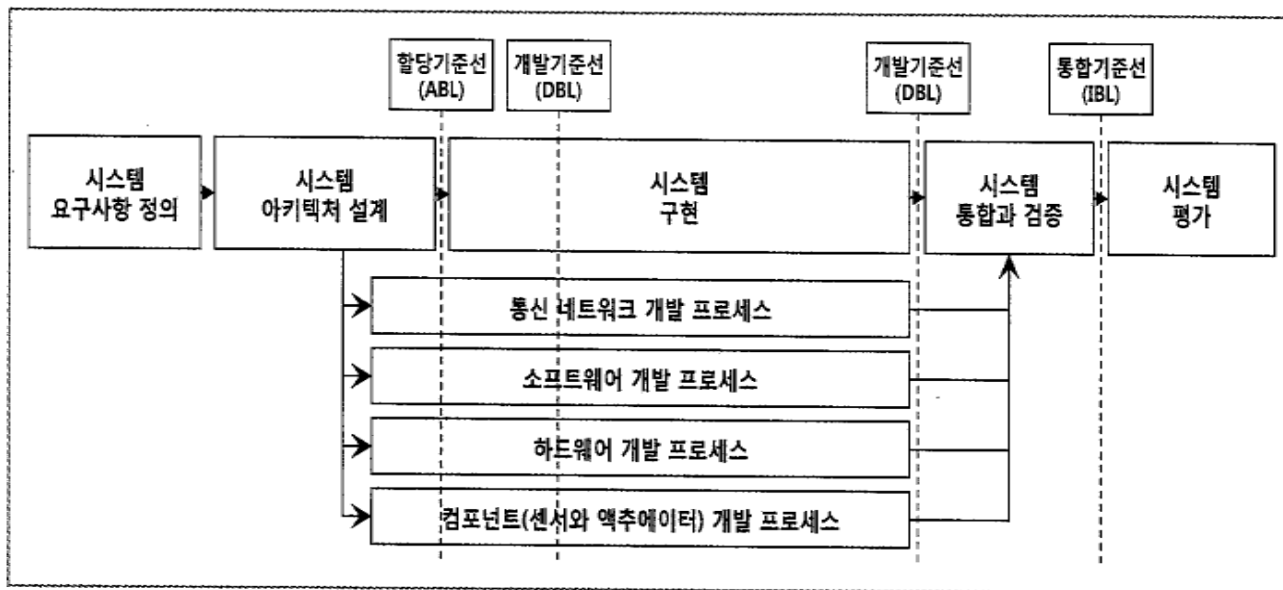
- 시스템 구성 요소간의 상호작용을 위한 내부 인터페이스, 송/수신 데이터 특성, 통신 방식 등 시스템 구성 설계
- 구성 요소간 전기적, 물리적 특성 등 시스템 구성 설계
- 동작 순서, 시간 조건 등 동작 개념 포함
- 시스템의 계층 구조에 따라 점진적으로 정의
- 고장과 위험 요소 식별하여 이를 감지하거나 인식,통제할 수 있는 기술적 해결책 정의
- 기술적 구현 가능성 검토 (**feasibility**)
- 엔지니어링 표준 규격 정의
- 제어 기술이 필요한 요소는 제어 알고리즘 설계



# System Implementation



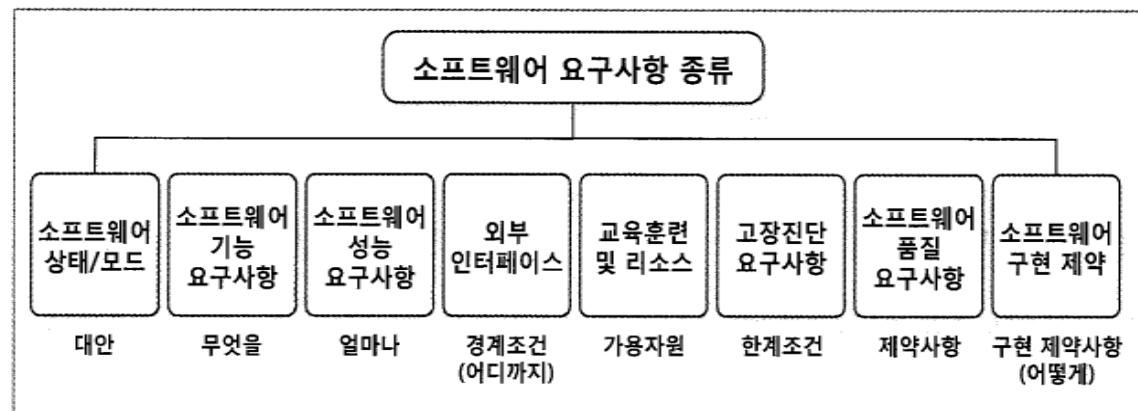
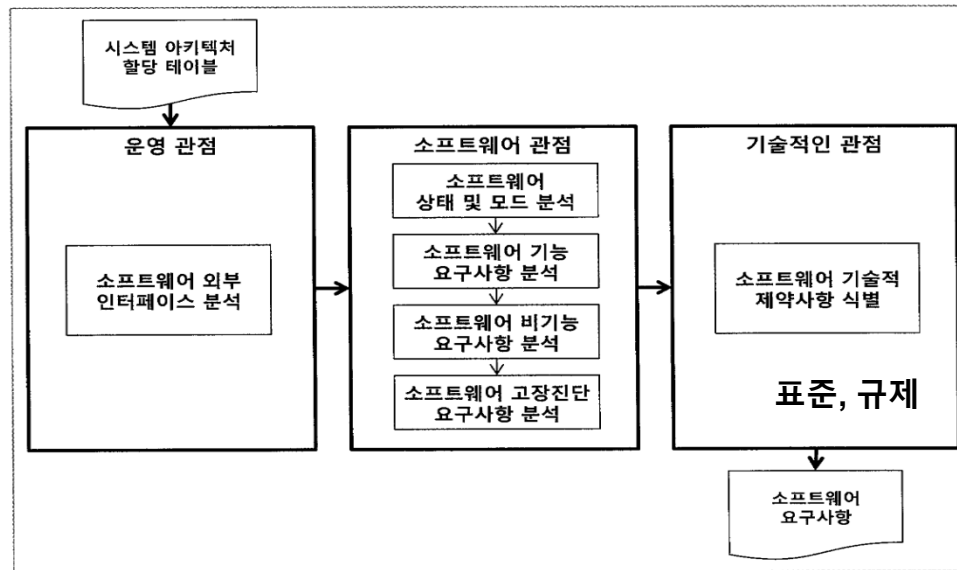
- ❖ 설계된 내용을 기반으로 시스템을 구체화하는 과정
  - Vehicle Network, S/W, H/W, Sensor, Actuator, etc
  - ABL : Allocation Baseline
  - BBL : Development Baseline ( Start & Integration )
  - IBL : Integration Baseline



# S/W Requirement Analysis



## ❖ 3 Steps Analysis



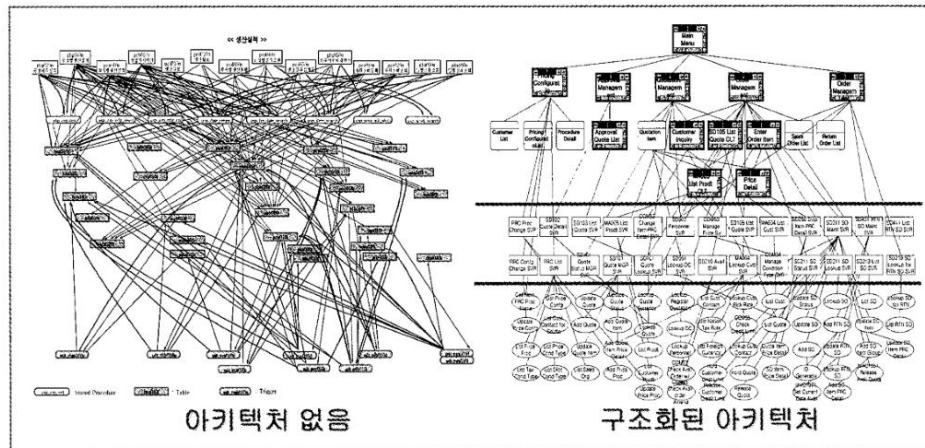




# S/W Architecture Design



- ❖ Embedded system에서 Architecture Design 중요
- ❖ 전체적인 S/W 구조를 추상화하여 표현
  - 선행 품질 확보, 제한된 Resource 확인
  - 기존적인 동작 예측 가능, 설계 변경에 대처 용이
  - Variant 차종에 재사용 고려



- ❖ 형상 관리 (SCM : S/W Configuration Management)





# S/W Architecture View



- ❖ “4+1 View Model’ proposed by Fruchten
  - 4 Design : Physical View, Implementation View, Process View, Logical View
  - 1 Verification : Use case view
- ❖ 기타 다양한 View Model 제안됨
- ❖ 여러 개발자들 간의 견해를 모의고 합의하기 위한 유용한 의사소통 도구
  - 이해 당사자들 간의 상충된 요구 사항 이해에 도움



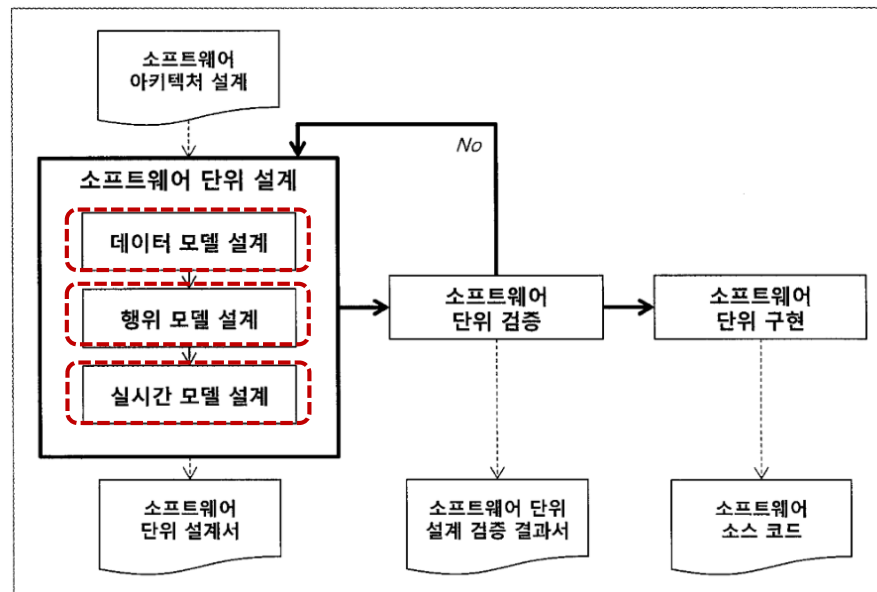


# S/W Unit Design & Implementation



- ❖ S/W Unit component의 요구사항과 I/F가 정의되고, 단위 기능별 상세 설계를 수행하고, 설계 내용에 따라 구현
- ❖ Basic S/W Layer : Firmware, OS dependent low level
- ❖ Application S/W Layer : High level
  - Model based auto code generation

- Data Model
- Behavior Model
- Real-time Model

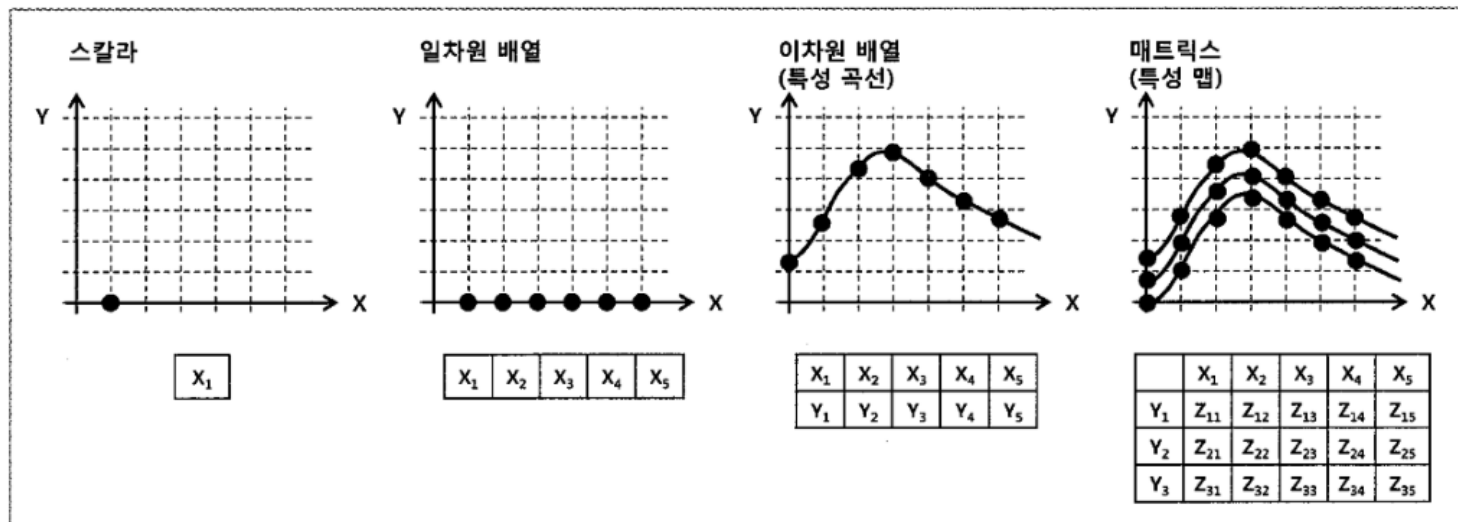




# Data Model

## ❖ Data Structure

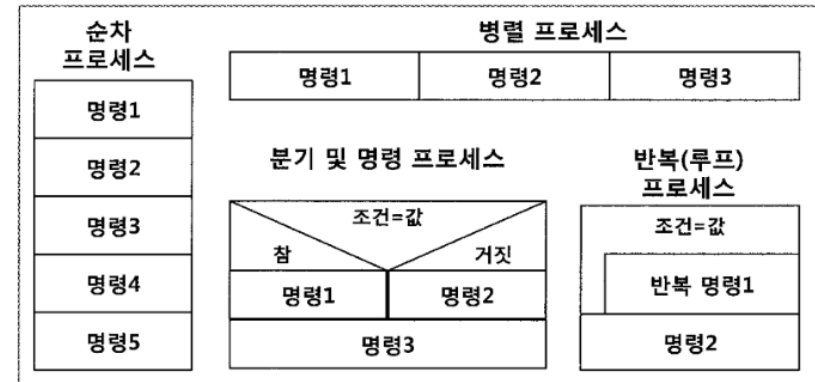
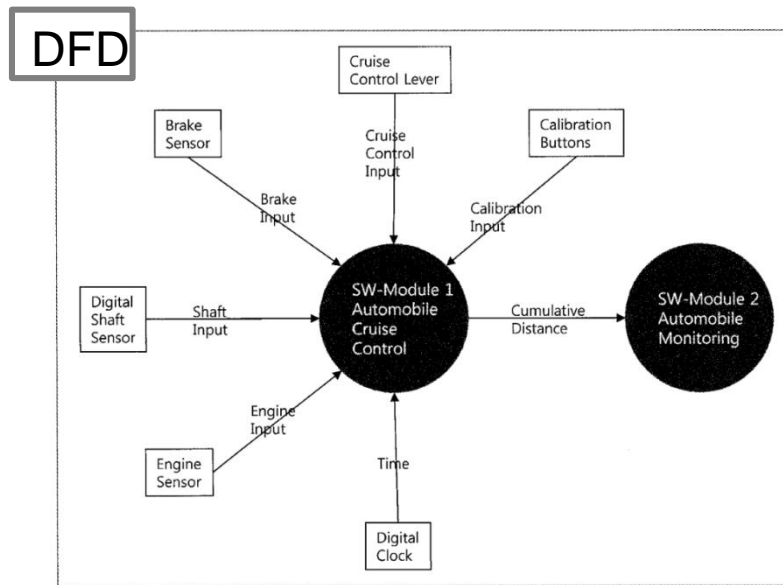
### ■ Ex) Gear Transmission Characteristic Map



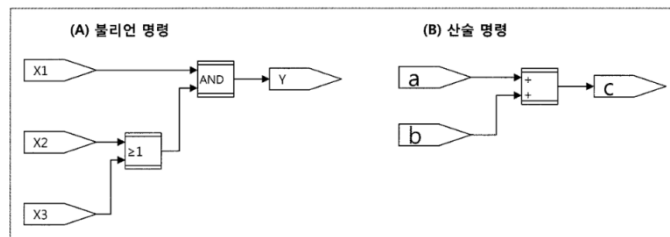


# Behavior Model

- ❖ Data Flow, Control Flow Diagram
- ❖ Design Tools : MATLAB, Simulink, ASCET



Nassi-Shneiderman Diagram :  
Control flow ( 명령, 실행 순서, 분기, 반복 )

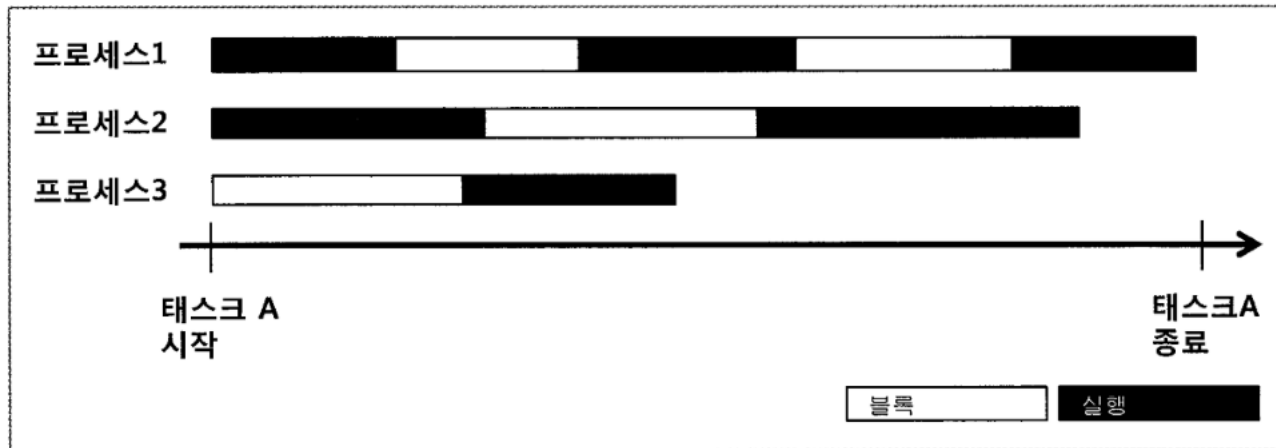




# Real-time Model



## ❖ Task Scheduling 이 가능한 OSEK OS 사용

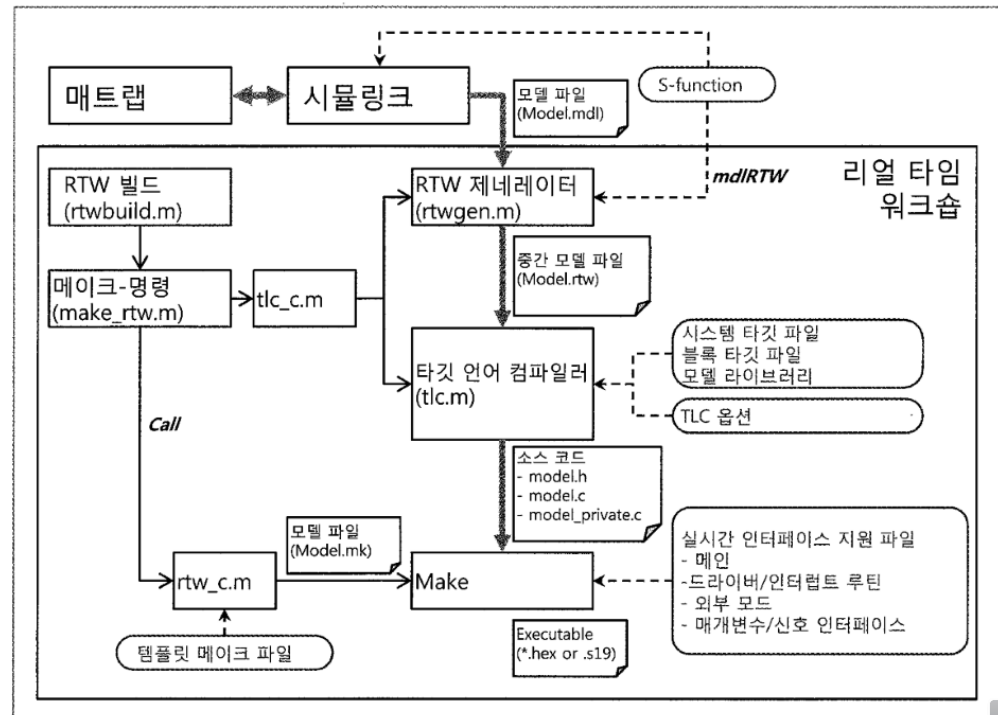
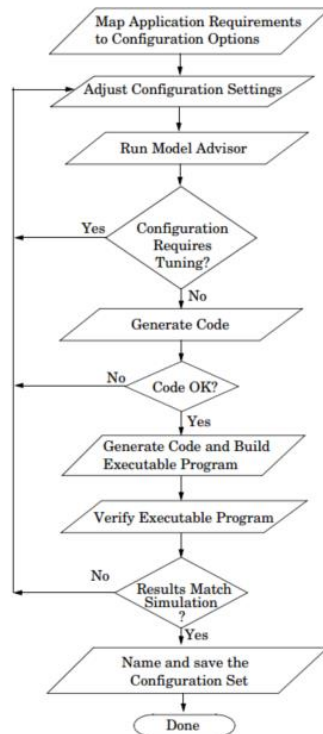




# S/W Unit Implementation

## ❖ Auto Code Generation

- Mathworks “*Real Time Workshop*”
- 개발자간 의사 소통, Human error 제거, Coding 시간 단축







# MISRA - C



## ❖ MISRA (Motor Industry Software Reliability Association)

- C coding guideline
- Readability, Maintenance, Reuse, etc
- 여러가지 **Code** 분석 도구가 있지만 **100%** 만족하는 도구는 없음  
(반드시 사람이 **Coding** 해야 하는 부분)
- **2012년 개정판 (유료) , 2004년 버전 (공개)**
  - Documented development process
  - Quality system capable of meeting the requirements of ISO 9001/ISO 90003/TickIT [11, 12, 13]
  - Project management
  - Configuration management
  - Hazard analysis
  - Requirements
  - Design
  - Coding
  - Verification
  - Validation
  - Training
  - Style guide
  - Compiler selection and validation
  - Checking tool validation
  - Metrics
  - Test coverage
- 전체 룰을 적용하면 개발자의 자유도가 제한돼 효율성이 떨어질 수 있으므로 조직 특성을 고려해 조기 표준 룰을 적용함

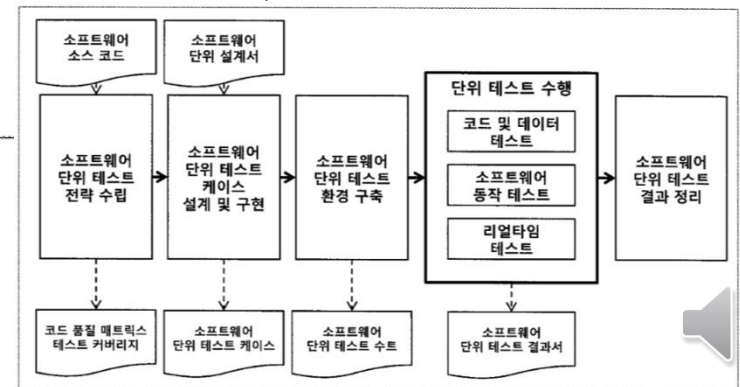
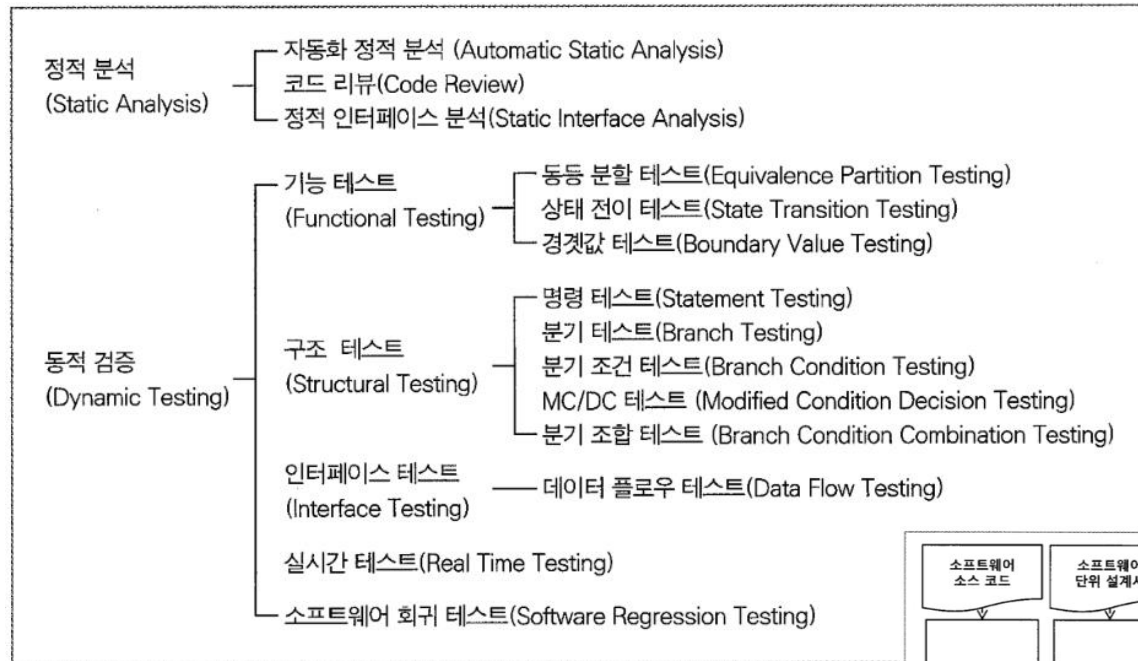


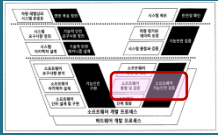




# S/W Unit Validation

❖ S/W unit가 정상적인 기능을 수행하는지 결함은 내포하는지 확인하는 과정 (**개발자 검증**)

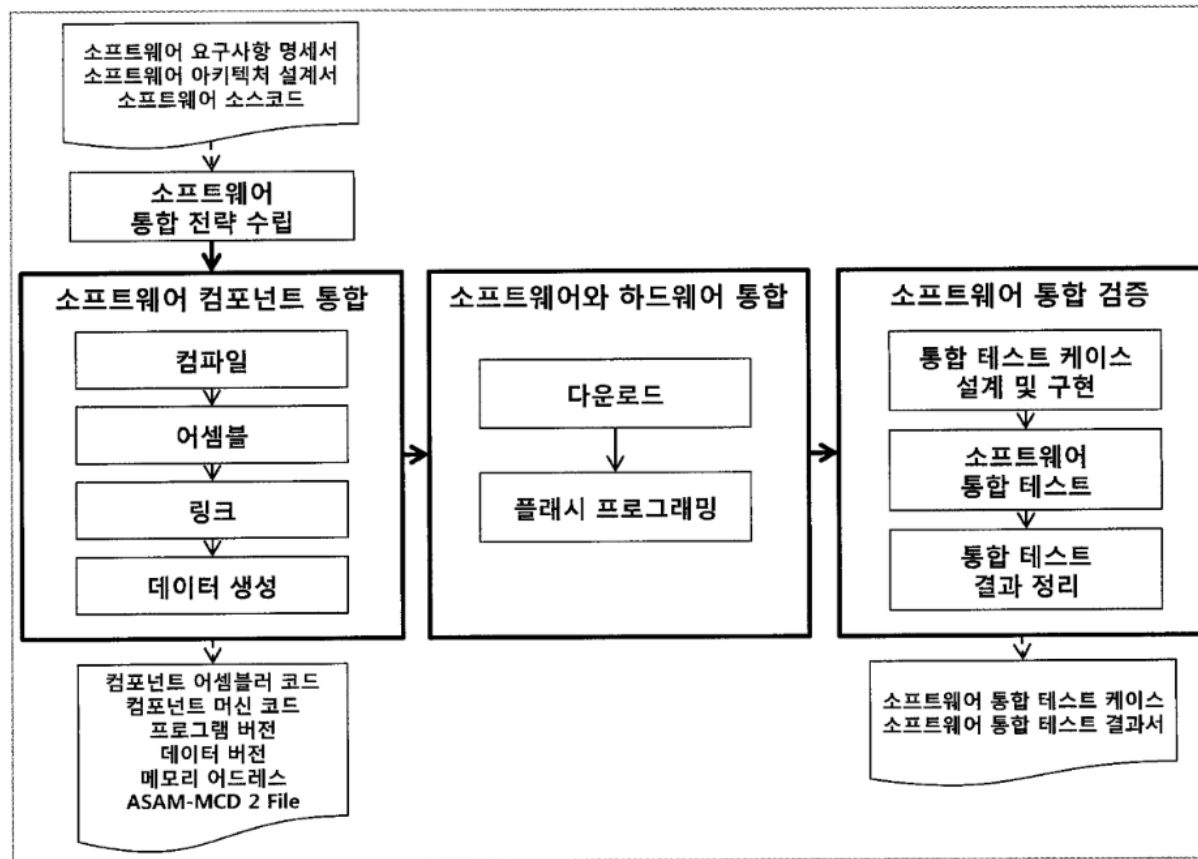




# S/W Integration & Validation



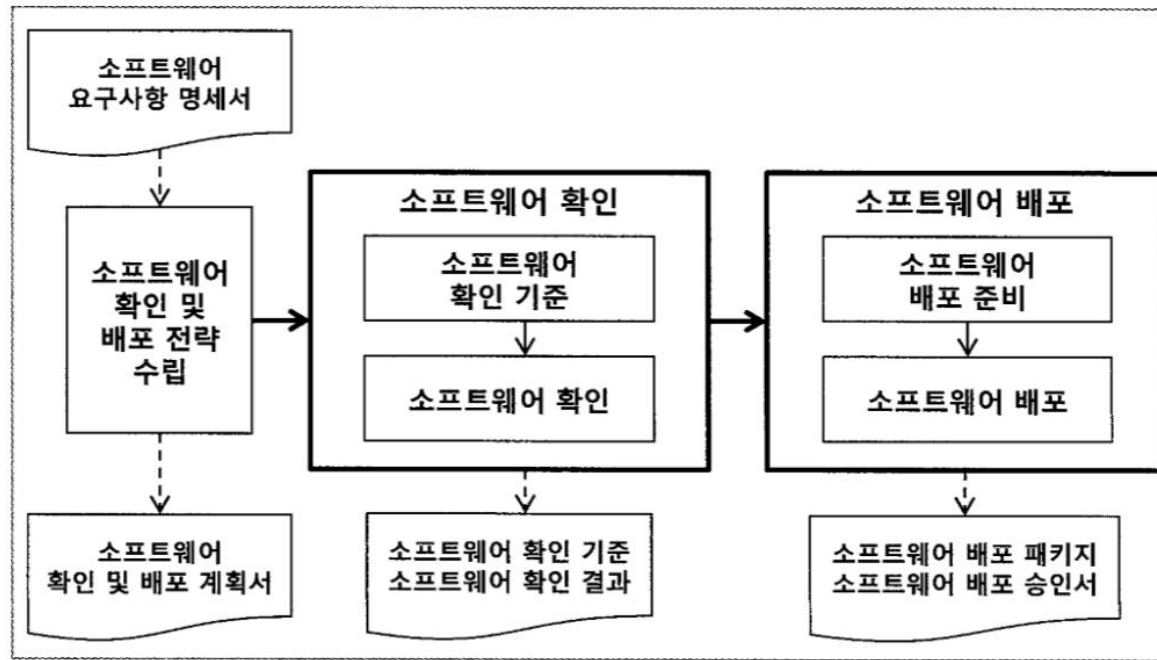
## ❖ S/W unit 간의 기능 의존성 검증(품질부서 검증)



# S/W Validation & Release



- ❖ ECU H/W 환경에서 요구사항 만족 여부 검증(품질부서 검증)
- ❖ 요구 사항을 모두 만족하면 배포

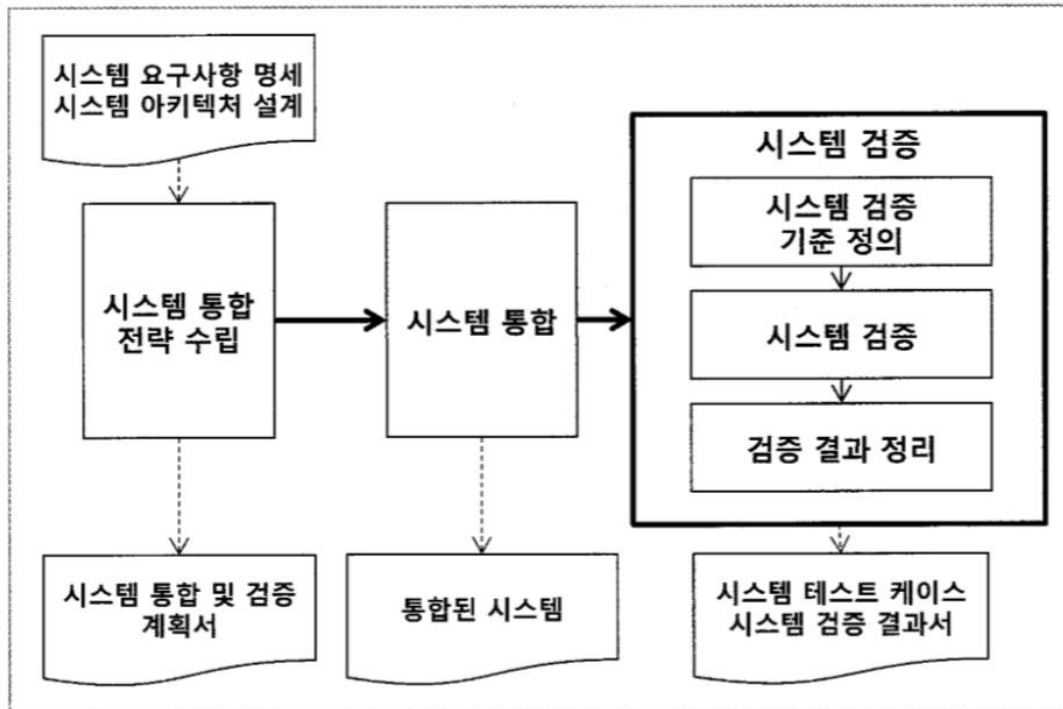


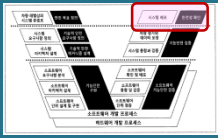


# System Integration & Validation



- ❖ 완성차 업체의 승인을 받고 협력업체들은 개발 시스템 납품
- ❖ 구현이 완료된 부품을 통합해 시스템(차량)이 요구하는 기능이나 성능을 만족하는지 통합 검증
- ❖ 하위 구성 요소 중에서 하나라도 미비하면 전체 시스템 개발 지연
  - 각 부품의 개발 일정을 준수하는 것이 중요

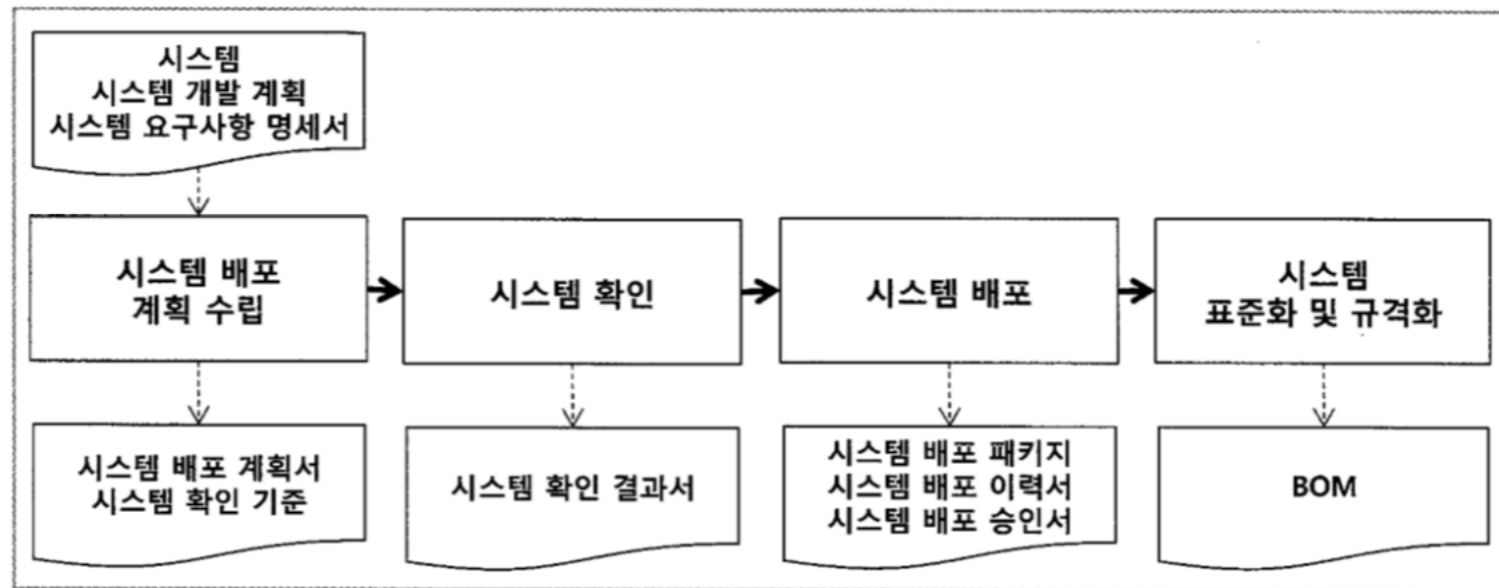




# System Release



- ❖ 차량에서의 기능 평가, **Data Calibration** 통해 성능 평가 수행하고 데이터를 확정하는 작업 (**완성차 업체 검증**)



\* BOM(Bill of Materials) 자재명세서



# Support & Management Process



- ❖ 제품의 품질과 신뢰성을 보장하기 위해서는 관리 중요
- ❖ 복잡한 시스템일수록 중요성 높아짐
- ❖ 관리 대상 : 품질, 비용, 일정, 인력, 기술, etc
- ❖ 관리의 영역

- 프로젝트 관리  
(의사소통, 진척, 위험, 이슈, ...)
- 요구사항 관리
- 형상 관리
- 변경 관리 (이력 관리)
- 품질 관리

|         |   |
|---------|---|
| 범위      | - 범위 정의 및 계획 수립<br>- WBS(Work Breakdown Structure) 생성<br>- 범위 검증 및 통제 |
| 시간      | - 활동 및 절차 정의<br>- 활동에 필요한 리소스 및 기간 산정<br>- 스케줄 정의 및 통제                |
| 원가      | - 비용 예측 및 예산 산정<br>- 비용 통제  |
| 품질      | - 품질 계획<br>- 품질 보증<br>- 품질 통제   |
| 인적자원    | - 인적 자원 계획<br>- 프로젝트 팀 구성(개발, 관리, 획득 관련 팀)<br>- 인적 자원 교육 훈련 및 관리 계획   |
| 의사소통    | - 의사소통 계획<br>- 이해당사자 관리<br>- 정보 배포<br>- 성능 보고                         |
| 위험 및 이슈 | - 초기 위험 식별 및 위험 관리 계획<br>- 정량적/정성적 위험 분석<br>- 위험 모니터링 및 통제            |
| 조달      | - 구입 및 획득 계획 수립<br>- 계약 계획 · 관리 · 종료                                  |
| 통합      | - 프로젝트 관리 계획<br>- 프로젝트 모니터링 및 통제<br>- 프로젝트 종료                         |





# Quality Management

- ❖ 프로젝트 수행기간 동안 최소의 노력과 비용으로 요구된 품질 수준에 도달할 수 있도록 개발 산출물에 대한 품질을 통제하고 보증하는 활동
  - **Development Life Cycle**
    - CMMI, Automotive SPICE
  - **Deliverables**
    - ISO 9126, ISO 14598



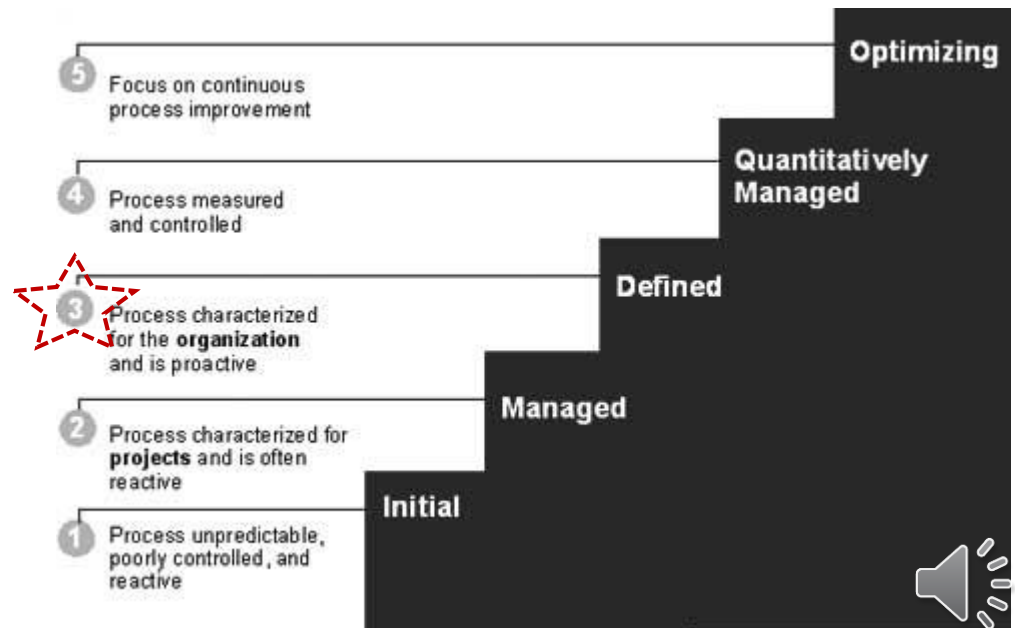


# CMMI



## ❖ CMM(Capability Maturity Model)

- 미 국방성은 우수 소프트웨어 개발업체를 객관적인 기준으로 선정하기 위해 1984년 카네기멜론대학에 SEI(Software Engineering Institute)를 설립
- SEI는 1991년에 SW 개발 조직이 높은 품질의 SW를 일관성 있고 예측 가능하게 생산하는 능력을 갖추도록 프로세스 모델인 SW-CMM을 개발
- SW-CMM은 SW 개발 능력 측정 기준과 SW 프로세스 평가 기준을 제공함으로써, 개발 조직의 성숙 수준을 평가할 수 있는 프로세스 모델로 개발
- 프로세스 : 일정한 목적을 위해 수행되는 일의 수행 과정을 의미하며 여기에는 필요한 역량, 인원 및 이를 지원하는 도구와 장비를 포함







# Automotive SPICE

## ❖ Automotive SPICE(Software Process Improvement and Capability Determination)

- The Automotive SPICE Process Assessment Model (PAM) has been developed by consensus of the car manufacturers within the Automotive Special Interest Group (SIG) of the joint Procurement Forum / SPICE User Group under the Automotive SPICE initiative. The Automotive SPICE Process Assessment Model (PAM) is available for use when performing conformant assessments of the software process capability of automotive suppliers in accordance with the requirements of ISO/IEC 15504-2. The Automotive SPICE Process Reference Model (PRM) is used in conjunction with the Automotive SPICE Process Assessment Model (PAM) when performing an assessment
- The Automotive SPICE Process Reference Model (PRM), which is defined in a separate document, is derived from Annex F and H of ISO/IEC 12207 AMD1: 2002 and ISO/IEC 12207 AMD2: 2004..

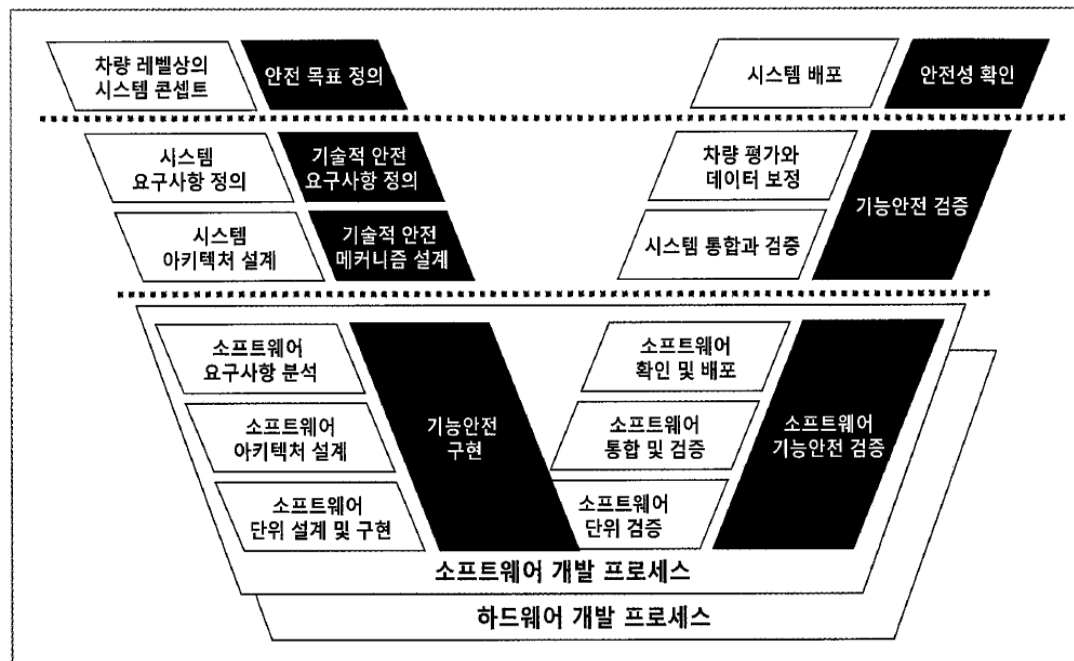




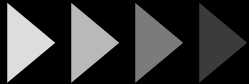
# Functional Safety

## ❖ Double V-Cycle for Functional Safety

- 차량의 안전을 확보하기 위한 Process
- ISO 26262
- ASIL (Automotive Safety Integrity Level) A ~ D



# Thank You !



충북대학교  
CHUNGBUK NATIONAL UNIVERSITY

