# AUTOSAR Communication stack

PDU -> 통신 종류에 상관없이 추상화, gateway 역할도 함

AUTOSAR COM -> 데이터를 쪼개서 하나의 field 조각으로 -> CANDB 정의된 시그널 기준으로 쪼개서 위로, 합쳐서 아래로 보내는 역할

CANDB -> CAN 설정 관련 데이터 (RPM과 같은, field 영역, 분수형을 위해 곱하는 등) 라잌 요구사항임

# NM: 일종의 절전모드

파워트레인 같은 거는 쓸 일이 없는데, BCM(바디 컨트롤 모듈)에서 많이 쓴다. Default가 sleep -> wakeup

VBAT -> sleep 모드에서도 CAN 신호를 받기 위해 전력을 공급하는 역할

여러 제어기들의 Sleep <-> Normal 모드를 제어하는 알고리즘이 바로 NM임. 가령, Master 제어기가 wakeup 이면 wakeup 하도록 or 모든 제어기가 wakeup 해야

OTA가 되는 MCU가 따로 있다 -> Flash가 두 개로, 실행중이 아닌 flash를 업데이트해서 실행중인 flash를 스위칭.

Flash 하나여도 실행중에 OTA를 할 수 있는 방법이 있는데, 시동이 꺼질 때까지 기다렸다가 하는 방식 -> 배터리를 사용해야하기 떄문에 방전 문제가 발생할 수 있음. 이때 NM 사용해서 방전 예방?

ASW -> Matlab stateflow, ASCET 많이 쓴다.

BSW -> C/C++

Matlab Simulink -> CAN 통신 해보래

MCU기반 플랫폼: 모빌진에 들어갈 AUTOSAR 블록들을 개발하는 듯?

AP기반 플랫폼: Adaptive AUTOSAR 뿐만 아니라 리눅스 기반 모든 거(진단 통신, CAN 통신), SOME/IP, 이더넷, 커널 컴파일(디바이스 드라이버) -> 리눅스(ex. 라즈베리)로 CAN 통신 빠르게 해본다거나 그런 프로젝트 해보면 좋을 듯, virtual machine(가상 환경 개발)

전동화 제어: 특정 제어기, 모빌진 뿐만 아니라 RTOS도 하나봐, A-SPICE 기반 산출물, 차량 시험하다가 BSW에 문제가 생기면 고치는 일

차량보안: MCU에 HSM이라는 코어 모듈이 따로 있다. 보안 수업 안들어서 별로일 듯. CAN은 암호화를 안하는데, 이더넷 계층은 암호화를 많이 한다. 양자학

설계도구: 모빌진 툴 개발, 코드 제너레이터 개발

프로젝트 : OTA, CAN 통신, RTOS 실시간

전력변환 소자 CDD (실시간 제어 개념 중요)