

진단통신(DoIP) 구현 목록

UDS

- SID (서비스 ID) + Parameter (데이터)
- 실제 UDS 프레임을 가져와서 일부를 사용할 예정
- 생각하는 SID 종류는 5개
 - 진단세션 (시작, 유지 등등)
 - 고유 정보 읽기 (ECU 정보, 라파 정보, 소프트웨어 버전 등등)
 - 센서 값 읽기 및 센서 고장 진단 (엔코더 2개, 초음파 3개, 레이저 1개)
 - 부품 강제 구동 (후미등 2개)
 - OTA 업데이트 (업데이트 실행, 업데이트 여부 등등)

구현 순서

1. 기반 기술 공부 및 테스트 (PC랑 라파 사이의 통신 테스트를 추가 수행해야 함)
2. 얼추 모든 테스트를 수행한 후에는 통신 프로토콜 설계 (맵핑 + UDS 조사)
3. 프로그램 구현

PC-라파 사이에서도 DoIP 메세지로 보내면

- 아키텍처가 깔끔해진다.
 - PC : 진단 메세지 생성하는 Tester
 - 라즈베리파이 : 메세지를 그대로 전달만 하는 Gateway
 - TC375 : 메세지를 처리하는 ECU
- PC에서 Wireshark로 데이터 패킷 확인 가능하다.
- 확장성이 좋다 (라즈베리파이 코드를 수정할 필요가 없다.)
- 실제 차량 진단 환경에서는 진단기가 DoIP 통신을 직접 수행한다.

페이지드 타입

- 0x0005 - 라우팅 활성화 요청
- 0x0006 - 라우팅 활성화 응답
- 0x8001 - 진단 요청 / 진단 데이터 보내줄 때도

- 0x8002 - 진단 응답 Positive Ack
- 0x8003 - 진단 응답 Negative Ack

Routing Activation → 이건 UDS가 아니고 DoIP (연결 확인용)

UDS 목록 (SID)

- 세션 활성 여부 확인 (0x3E)
- 데이터 읽어오기 (0x22)
 - 초음파, ToF, 엔코더
- DTC 읽기 (0x19) → 고장 메시지 읽기 (고장 데이터 읽기)
- DTC 설정 (0x85) → 고장을 기록할지 말지 설정
- DTC 소거 (0x14) → 고장 기록 삭제 (더 깊게 파고들면 특정 고장 기록만 지우고, 전부 다 지우고 아마 우리 도 전부 다 삭제)

→ 해당 요청에 대해 긍정 응답일 때는 0x40을 더해서 함 (긍정적인 응답을 할 때 규칙 / 0x22만이 아니라 긍정응답은 무조건 SID에 + 0x40 그리고 뒤에 필요에 따라 데이터 붙이고)

예시: 페이로드 탑입 0x8001로 데이터 읽어오기 SID 0x22 모터 DID (모터 데이터 가져와라)

→ 응답할 때 페이로드 탑입 0x8001, SID $0x22+0x40 = 0x62$, 모터 DID + 요청받은 데이터를 붙임 (페이로드 길이가 늘어나겠지)