**7주차 업무 일지**

**결론**

MCU와 Jetson의 SPI 통신 테스트 완료

RC카 설계 및 제작

RC카 제어 알고리즘 및 송수신 데이터 설계

깃허브 리팩토링

**손주석(팀장, RC카 제작 및 제어 알고리즘 구현)**

**내용**

**RC카 설계 및 제작**

1. RC카 부품 변경에 따른 설계 조정

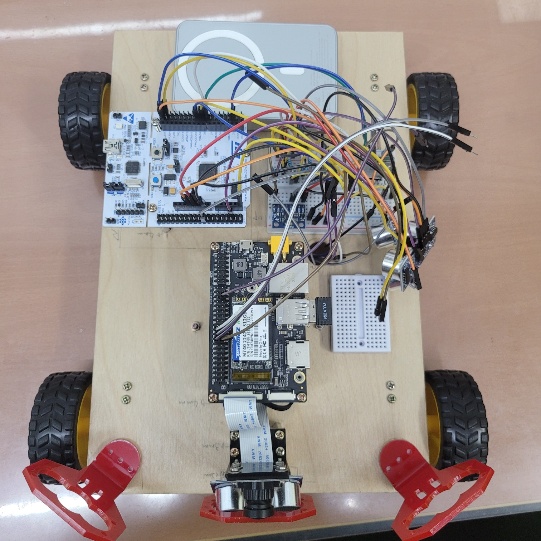
i) RADAR 미사용 결정에 따른 초음파 센서 설계 추가

ii) 차체 안정성을 위한 DC모터 위치 조정

2. RC카 제작

i) 설계도에 따른 부품 조립 및 오차 수정

ii) 제작한 RC카를 기존 코드를 이용해 테스트 진행



**MCU와 Jetson SPI 통신 설정**

1. 핀 연결

i) Jetson을 Master, MCU를 Slave로 설정해 MISO, MOSI, SCK, CS핀을 연결함. 추가적으로 GND 핀도 상호간 연결

2. 수동 CS 설정 -> 실패

i) MCU 설정 - GPIO핀에 NSS 설정을 하지 않고 GPIO input을 설정함으로써 Jetson이 통신 유무를 제어하도록 함

ii)Jetson 설정 – Jetson 측에서 CS핀의 통신을 GPIO를 이용해 관리하도록 코드를 작성하고 실행함. 하지만 GPIO 설정에 대한 허용 권한이 없고 코드가 복잡해져 다른 방안을 이용하는 것으로 결정함.

3. 자동 CS 설정(NSS)

i) MCU 설정 – ioc의 GPIO핀에 NSS 설정을 이용함에 따라 CS핀이 자동으로 관리되도록 설정함.

ii) Jetson 설정 – Raspberry pi에서 SPI 통신으로 쓰던 예제와 동일하게 자동으로 CS핀이 관리되도록 함.

**MCU의 모터 드라이버 오류 수정**

1. 센서 전류 과다 사용으로 예상한 디버깅 진행

i) IMU, 초음파, Cds 센서 등을 사용하지 않고 모터드라이버와 블루투스만을 이용해 작동하는 코드를 이용해 테스트를 진행함.

ii) 다른 센서를 사용하지 않음에도 동일하게 동작이 되지 않는 것을 확인함.

2. 오실로스코프를 통한 전압 분석

i) 모터 작동 신호를 관리하는 핀인 ENA와 ENB핀을 확인함. 하지만 파형 및 전압에 문제가 없는 것을 확인함.

ii) IN1 ~ IN4 핀 전압을 확인함. 파형이 디지털형태가 아닌 아날로그형태가 나오는 것을 확인하였으며 전압이 의도한 바와 다르게 낮게 걸리는 것을 확인함.

3. 핀 연결 수정

i) GND핀에 다른 센서가 너무 많이 연결되어 있어 간섭을 고려해 서지완 학생이 GND 핀을 수정함. 후진을 제외한 나머지 동작이 문제없이 작동하는 것을 확인

ii) IN1 ~ IN4 핀 수정. 현 IN핀에 다른 인접한 핀이 없도록 만들기 위해서 핀의 번호를 수정함. IN핀을 다른 핀들과 격리함에 따라 신호 간섭이 사라져서 후진 동작이 문제없이 작동하는 것을 확인할 수 있었음.

**Github refactoring**

1. 필요없는 디렉토리 삭제

2. MCU와 Jetson에서 각각 git clone을 통해 작업을 진행할 수 있도록 분리

3. 이외 Docs와 Test에 대한 디렉토리 정리

**계획**

RC카 제어 알고리즘의 설계에 대한 구체화 진행

MCU와 Jetson 데이터 송수신에 대한 설계 구체화 및 구현

주요 설계 알고리즘을 테스트

문제가 발생하는 알고리즘 디버깅 및 테스트 진행