

배달 안전성을 위한 흔들림 방지용 모바일로봇 개발

수행분야	졸업작품제작	
지도교수	김종형 교수님	확인
팀원	15181434	이인호
	16100170	이동건
	17100139	양성백
	19100054	김시현



기계시스템디자인공학과

목 차

가. 서론

- 가.1. 개발 배경 (주제 선정 동기)
- 가.2. 사례 연구
- 가.3. 목적 및 의의

나. 작품의 설계

- 나.1 Stabilizer 설계
- 나.1.1. 이론적 배경(양성백)
- 나.1.2. 최적화(양성백)
- 나.1.3. 모델링(양성백)
- 나.2. 서스펜션 설계.....
- 나.2.1. 이론적 배경(양성백)
- 나.2.2. 서스펜션 모델링 및 동적해석(이인호).....
- 나.2.3. 수학적 모델링 및 시뮬레이션(양성백) ...
- 나.2.4. 최적화(양성백)
- 나.2.5. 실험(양성백)
- 나.3. 하위 제어(모터, 컨트롤러 등).....
- 나.3.1. Stabilizer 제어(이동건) ...
- 나.3.1.1. 모터 선정 및 개발 과정(이동건)....
- 나.3.2. AGV DC 모터 제어(김시현)
- 나.3.3. IMU 센싱 및 필터링 구성(이동건)
- 나.4. 협조 제어 알고리즘
- 나.4.1. 속도 프로파일 구성(김시현)
- 나.4.2. 유체 제어 알고리즘 구성 및 실험(이동건)

나.4.3. ROS를 활용한 상위통합 명령제어(김시현)

나.5. 제어부 및 회로설계

나.5.1 AGV 및 Stabilizer 회로도

나.5.2 주요 부품

다. 결론 및 향후과제

다.1. 실험 결과.....

다.2. 기대 효과.....

다.3. 작품의 한계 및 보완.....

라. 여름방학 그룹 스터디

라.1. 이인호 그룹스터디.....

라.2. 이동건 그룹스터디.....

라.3. 양성백 그룹스터디.....

라.4. 김시현 그룹스터디.....

라.4.1. 아두이SH

라.4.2. ROS

부록

A. 논문 및 참고문헌

B. 참고 사이트

C. 개인별 Study

가. 서론

가.1 개발 배경 (주제 선정 동기)

COVID-19 바이러스의 등장으로 세계의 경제가 큰 위기에 빠지며 직접적 접촉없이 소비가 가능하고, 공간과 비용 측면에서 강점을 지닌 무인화 기술이 오프라인 유통매장을 중심으로 빠르게 확산되었다. 편의점, 카페, 세탁소 등의 점포에서 무인 시스템을 도입한 모습을 발견할 수 있다.

이러한 무인화 흐름은 요식업계에도 적용되고 있다. 이미 많은 기업들과 로봇 스타트업 업체들은 서빙 로봇 시장에 뛰어들었고, 외식과 호텔 업계에 로봇 도입이 빠르게 확산되고 있다. 서빙 로봇은 단순히 직원 1명을 대체하는 것이 아니라 다양한 기능을 보유하여 다역의 역할이 가능하기 때문에 인건비 부담을 크게 줄일 수 있는 좋은 수단이다.

그러나 아직도 서빙 로봇의 한계는 존재한다. 찌개류나 국물 등의 음식이 많은 우리나라에서는 이러한 음식을 이송하기 위해서는 의도적으로 로봇의 속도를 줄여서 음식이 흘러넘치지 않게 조절해야한다. 또한 음료 등을 이송할 때에도 마찬가지로 여서 카페에서 서빙 로봇을 사용할 때는 뚜껑이 있는 테이크 아웃잔을 사용하여 음료의 흘러넘치는 것을 막거나, 로봇의 속도를 줄여야만 한다. 대형 음식점이나 카페에서는 너무 느린 로봇의 속도는 일 효율성을 감소시킨다. 배달 로봇 또한 이미 로봇 서비스 시장에 뛰어들었으며 실제 서비스하고 있는 배달 업체도 존재한다. 그러나 서빙 로봇보다 더 불안정한 환경과 노면을 더 빠른 속도로 주행해야하는 배달 로봇의 사정도 마찬가지로, 뚜껑과 포장으로 단단히 방어되고 있는 음료라도 울퉁불퉁한 노면을 주행하며 흘러넘치거나 엉망이 되고 만다. 만약 이러한 흘러넘침의 걱정 없이 음식과 음료를 이송할 수 있는 시스템이 있다면 서빙 로봇은 한 층 더 향상되어 효율이 증가할 것이다. 우리는 이러한 문제를 해결하기 위하여 흘러넘침 없이 안정적으로 식음료를 이송할 수 있는 모바일 로봇을 개발하고자 한다.



그림 1-1 배달 로봇과 배달된 음료의 모습, tvN 방송 캡처

가.2. 작품 개요

실내 매장의 모바일 로봇의 주행 중에 생기는 가속도로 인한 tray의 흔들림으로 유체수면에 기울기가 생긴다. 기울기가 커질 경우 유체의 흘러 넘침이 발생하게 되는데, 이때 트레이의 각도를 보정하여 흘러 넘침을 방지해준다. Stabilizer의 tray를 가속도 프로파일에 따른 보정각도로 제어하고, 동시에 모바일로봇의 속도프로파일에 따른 가감속 제어를 통하여, 빠르고 흔들리지 않도록 음료를 배송한다.

나. 작품의 설계

나.1. Stabilizer

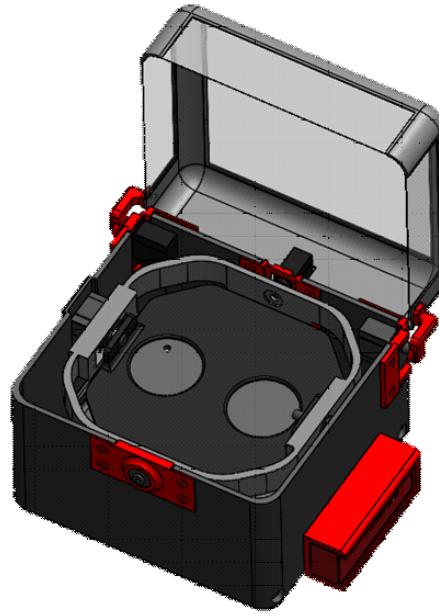
나.1.1. 이론적 배경

작년 작품, 2자유도 그네 구조의 단점은 plate의 크기가 작고 허용 하중량이 적다는 점이다. 그 원인은 크게 2가지, plate의 무게 중심 문제와 그네 구조의 구조적 문제이다. 그네 구조는 모터 축을 기준으로 움직인다. 무게 중심 위치가 작동 축으로부터 멀어진다면 그리고 그 크기가 증가한다면 모터 토크는 급격하게 증가한다. 따라서 무게 중심을 고려한 모터 토크는 급격히 증가한다.

무게 중심을 고려했을 때 롤 작동 모터는 높은 토크가 필요하고 그에 따라 크기와 무게도 증가한다. 무게가 증가하면 피치작동 모터는 롤 작동 모터의 무게와 plate의 무게중심으로 인해 더욱더 큰 토크를 요구한다. 크기가 증가하게 된다면 그네 구조의 프레임은 전체적으로 커지게 된다. 즉 2자유도 그네 구조는 모터의 토크를 증가시키는 것에 한계가 있다. 따라서 2자유도 그네 구조는 plate의 크기를 감소시켜 무게 중심의 이동을 제한을 시키거나 허용 하중량을 낮춰야 한다.

반면 병렬 구조는 여러 개의 모터가 plate밑에 위치함으로 plate의 무게 중심의 변화로부터 토크 영향이 보다 적다. 또한 구조적으로 plate의 크기에 영향을 받지 않는다. 병렬 구조는 그네 구조보다 적정 하중량이 더 크고 plate의 크기를 더 크게 할 수 있다.

위와 같은 이유로 병렬 구조를 사용하여 작년 작품과 동일한 공간을 사용하며, plate의 크기와 plate의 무게 중심을 고려한 적정 하중량을 증가시키고 했다.



<작년 작품, 2자유도 그네 구조>

여러 병렬 메커니즘 중 Fig. 1와 같은 3-RRS 메커니즘[1]을사용했다.

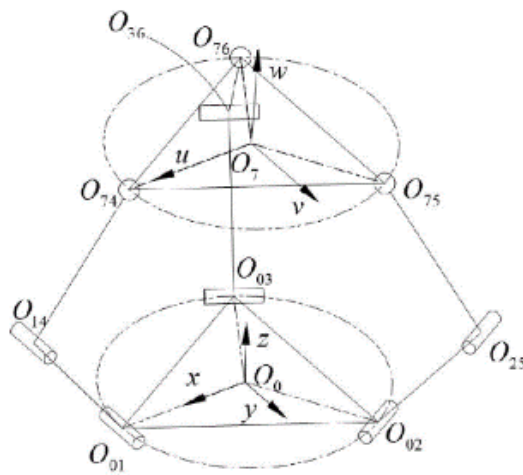


Fig. 1 3-RRS Parallel mechanism

나.1.2. 최적화

한정된 공간 내에서 Stabilizer의 Roll, Pitch값은 적어도 17도 이상이어야 한다. 이를 만족하는 설계 변수들에 대해 최적화 계산을 했다. 도구는 Matlab Global Optimization Toolbox[2]를 사용하였으며, 알고리즘은 GA를 사용했다. Optimization Toolbox 사용법에 대해서는 첨부자료 [B]-1 220712study편을 참고하

면 된다(Global Optimization과 Optimization은 사용법 동일하다).

나.1.2-1 최적화 문제정의

Stabilizer의 설계 변수들은 Fig. 2와 같다. 설계 변수들은 D, d, link1, link2, H이다. D, d, link1은 제한된 공간으로 인해 고정된 값이다. 각각 D = 110(mm), d = 85(mm), link1 = 22.5(mm)이다. Link2, H는 최적화 변수이다.

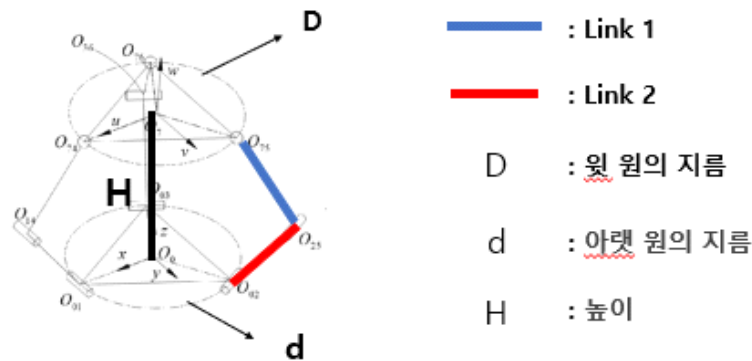


Fig. 2

최적화 과정[3]은아래와같다.

- 1) Pitch를 x축, Roll을 y축으로 설정. Stabilizer의 잠재적 영역($-40 < x < 40$, $-40 < y < 40$)을 설정.
- 2) 잠재적 영역 내부에 임의의 점들을 선정하고 그 점들의 총 개수를 N_total로 정의.
- 3) 모든 임의의 점들을 역기구학 문제에 대입하여 ρ값을 얻는다. 이 때 공간, 조인트의 구속 조건을 고려했을 시 ρ값이 조건 내에 있는지 판단한다.
- 4) 조건 내에 있는 점들의 개수를 N_in으로 정의
- 5)영역을 최대화하기 위해 N_in과 N_total의 비, W를 계산하여 가장 큰 값을 찾는다.

$$W = \frac{N_{in}}{N_{total}}$$

또한 모든 방향으로 같은 각도로 움직일 수 있어야함으로 작업영역의 모양은 원[3]으로제한하였다.

따라서 최적화 문제 정의는 다음과 같다.

Maximize $f(x) = W$

over $x = [\text{link2}, H]$

subject to $g_1 : [30, 30] < x < [100, 100]$

나.1.2-2 최적화 결과

최적화 결과 $link2 = 60(mm)$, $H = 59(mm)$ 이다.

이때의 작업영역은 Fig. 3와 같다. 최대 Roll, Pitch 값은 약 20도이다.

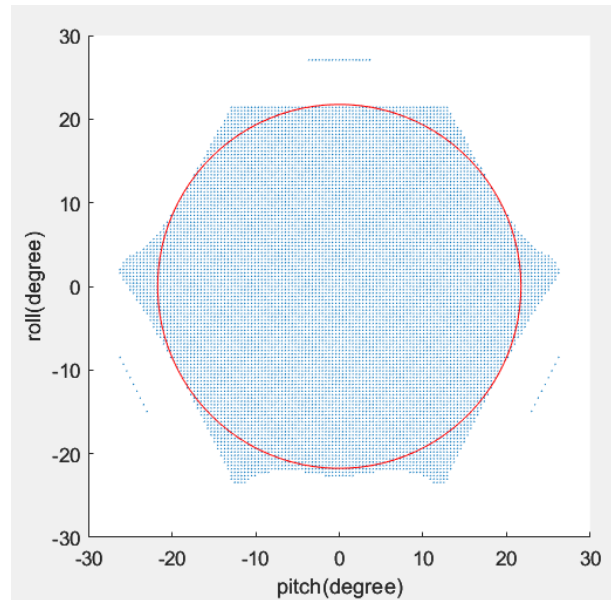


Fig. 3 Stabilizer 작업영역

나.1.3 모델링

나.1.3-1 Stabilizer 모델링 및 제작

최적화 계산을 통해 얻은 설계 변수 값들을 적용하여 Fig. 4와 같이 모델링 및 Fig. 5와 같이 제작하였다.

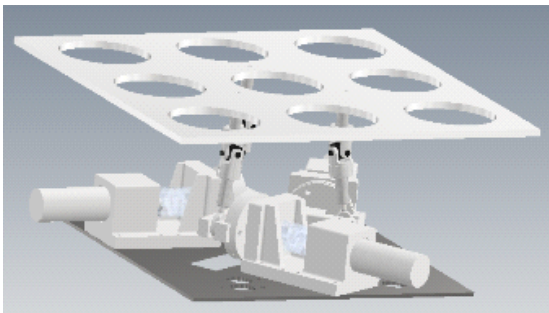


Fig. 4

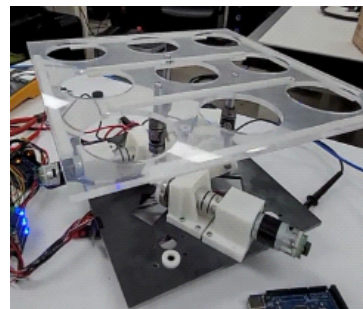


Fig. 5

나.1.3-2 Stabilizer 병렬 구조, 그네 구조 토크 비교

무게 중심 위치에 따른 토크를 비교하였다. 무게 중심의 크기는 2Kgf, 3Kgf이다. 무게 중심의 위치는 Roll축 위에 존재하며 원점으로부터 35mm, 70mm, 105mm에 위치한다. Plate는 Pitch각도로 움직이며 Fig. 7의 프로파일로 움직인다. 작동 시의 최대 토크를 분석하였고 이때 Inventor dynamic simulation을 사용했다.

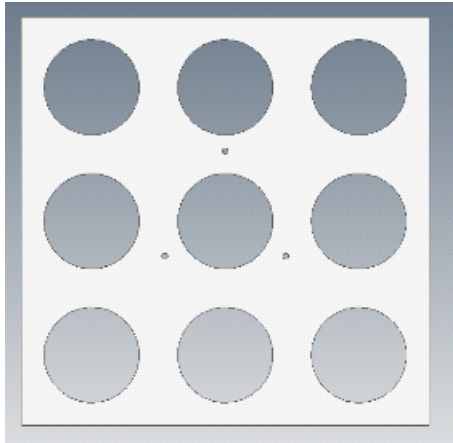


Fig. 6 Plate

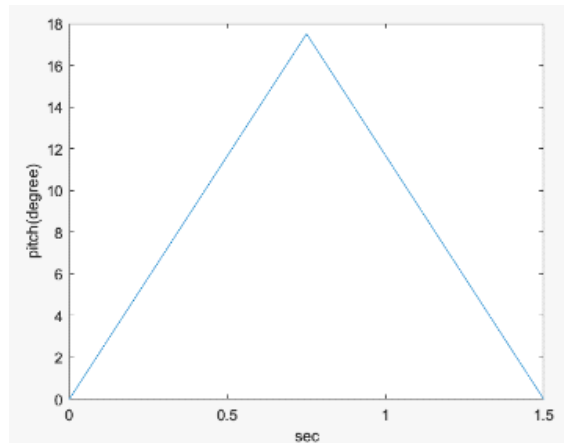


Fig. 7 Profile

Table. 1~4는 토크 분석한 결과이다. 토크의 단위는 N*mm이다.

그네 구조는 무게 중심 위치에 따라 토크가 급격히 변함을 알 수 있다. 그에 반에 병렬 구조는 변화량이 적음을 알 수 있다.

무게 중심의 크기가 변화함에 따라 그네 구조의 토크 변화량은 큰 폭으로 증가하였다. 반면 병렬 구조는 적은 폭으로 증가하였다.

그네 구조는 무게 중심의 크기, 위치에 따라 토크가 크게 바뀔 수 있다.

2 Kgf 병렬	무게 중심 위치		
	35(mm)	70(mm)	105(mm)
Motor 1	306.22	466.82	627.415
Motor 2	-15.327	-175.93	-336.529
Motor 3	146.22	146.22	146.22

Table. 1

2 Kgf 그네	무게 중심 위치		
	35(mm)	70(mm)	105(mm)
Motor	686.7	1373.4	2060.1

Table. 2

3 Kgf 병렬	무게 중심 위치		
	35(mm)	70(mm)	105(mm)
Motor 1	459.321	700.214	941.123
Motor 2	-23.098	-263.99	-504.794
Motor 3	219.44	219.33	219.33

Table. 3

3 Kgf 그네	무게 중심 위치		
	35(mm)	70(mm)	105(mm)
Motor	1030.05	2060.1	3090.15

Table. 4

나.2. 서스펜션 설계

나.2.1 서스펜션 모델링 및 시뮬레이션

작품의 전체 서스펜션에 대한 수학적 모델링은 Half-suspension model로 진행했다. 이는 Fig. 8과 같다. 이 모델은 Stabilizer의 무게(m), AGV의 무게(M), 캐스터 바퀴의 무게(m_f, m_r), 모터 바퀴의 무게($wheel$)로 구성되어 있다.

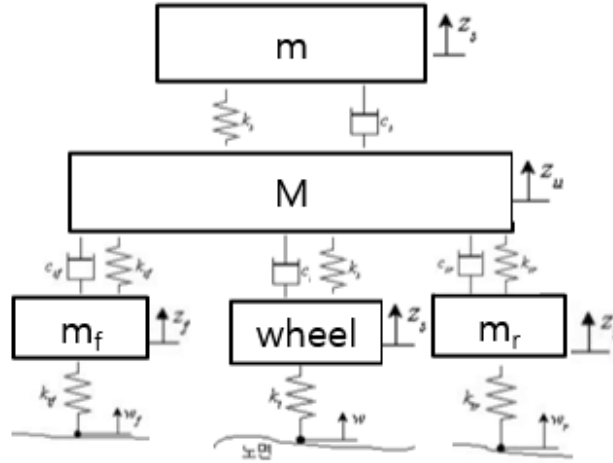


Fig. 8 Suspension-system

Fig. 9의 장애물을 1m/s로 주행하는 것에 대해 시뮬레이션을 구성했으며 프로그램은 매트랩을 사용하였다. Table. 5에 대하여 시뮬레이션을 진행했을 경우 Stabilizer에 발생하는 z축 가속도 그래프는 Fig. 10와 같다. 이에 대한 코드는 [B]-2에 첨부했다.

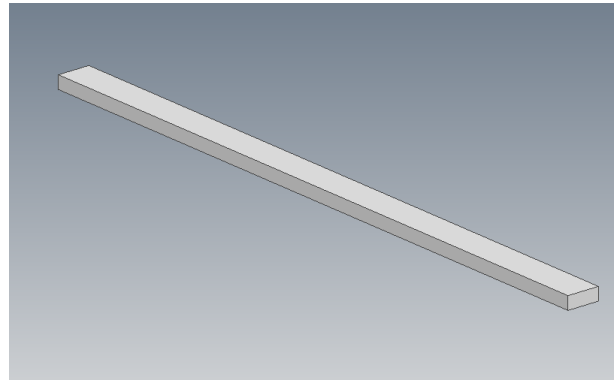


Fig. 9 높이 7mm, 너비 10mm의 장애물

$m = 8.5(\text{kg})$	$K_{sf}=40(\text{N/mm})$	$C_{sf}=0$
$M = 13.5(\text{kg})$	$K_s=40(\text{N/mm})$	$C_u=500(\text{N*m/s})$
$m_f=0.1(\text{kg})$	$K_{sr}=40(\text{N/mm})$	$C_d=500(\text{N*m/s})$
$wheel = 0.1(\text{kg})$	$K_{wf}=200(\text{N/mm})$	$C_{wf}=0$
	$K_{wr}=200(\text{N/mm})$	$C_{wr}=0$

Table. 5 설계 파라미터 예시

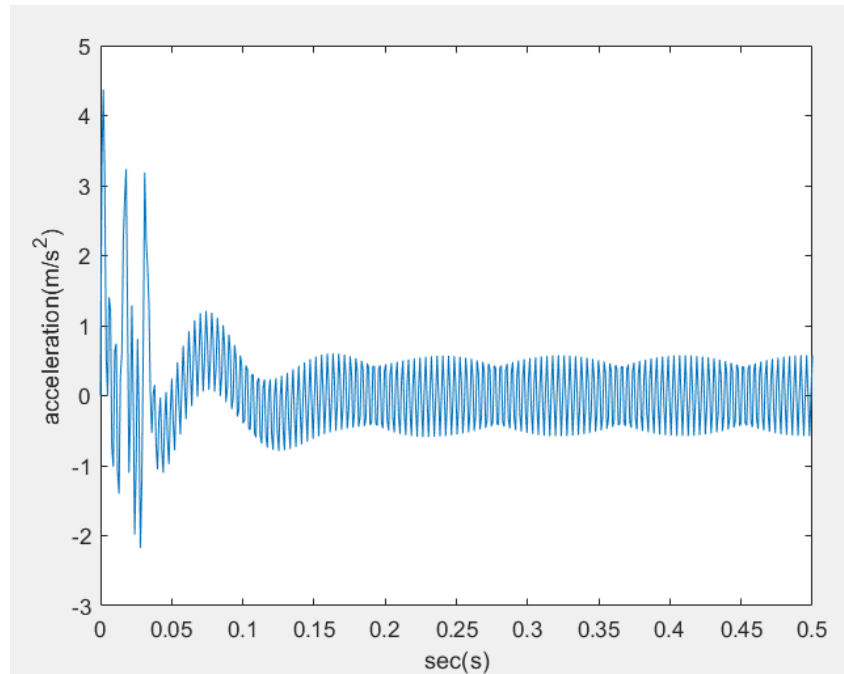
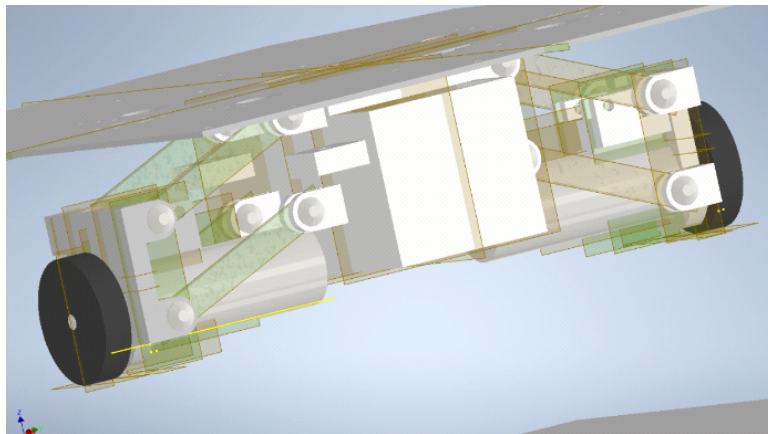


Fig. 9 Stabilizer에 발생하는 z축 가속도 시뮬레이션 예시.

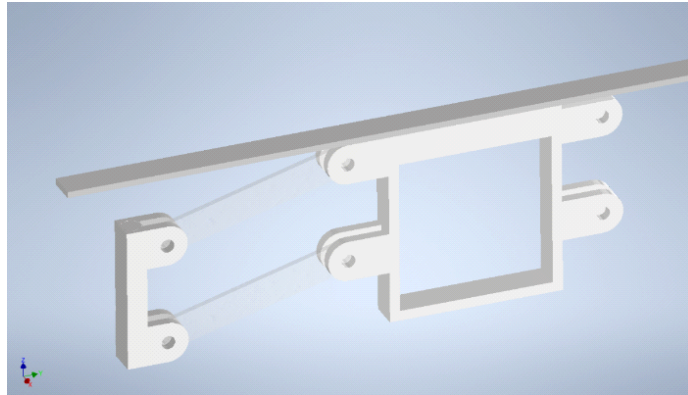
나.2.2. 현가장치 모델링 및 동역학적 해석을 통한 강성 확인

AGV 현가장치의 구조는 Original Suspension of the AGV ActiveOne XL(3)을 적용했다.



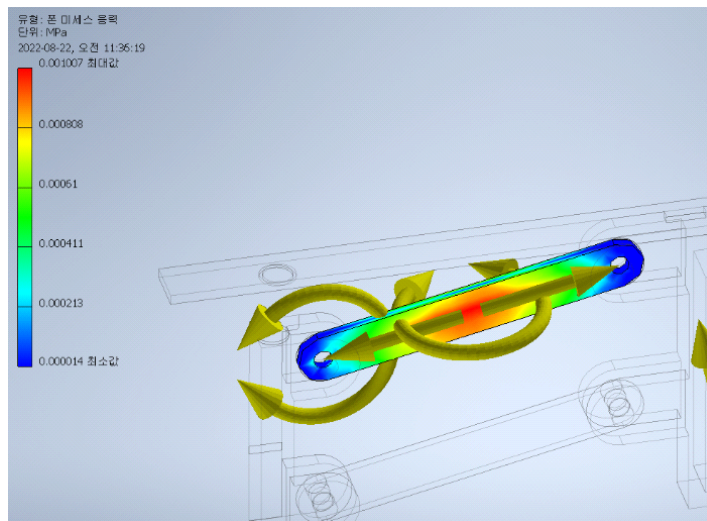
<현가장치 모델링>

AGV 설계요구조건은 현가장치는 주행을 위해 충분한 강성을 가져야 하는 점이다. 동적 해석은 정적 해석으로는 알 수 없는 정보를 알아볼 수 있다. 그래서 이를 알아보기 위해 해석을 진행했다.



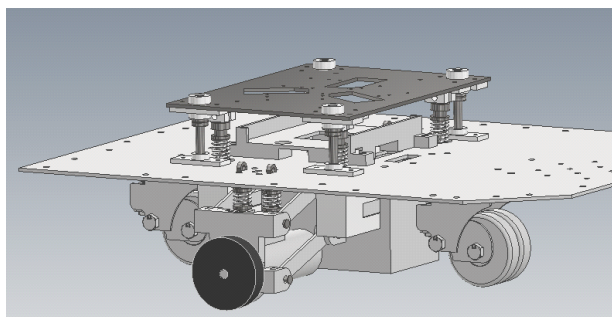
<현가장치 형태 최소화>

강성, 자유길이, 댐핑은 AGV 제원에 맞게 수치를 넣었고, 받는 힘은 장애물 주행 실험에서 얻은 데이터를 이용했다.



<모터링크 응력분포 모습>

AGV에서 가장 취약한 부품은 모터링크다. 모터링크의 응력, 변형률을 확인한 결과 AGV 주행하는 데에 있어서 충분한 강성이 있는 모습을 확인했다. 해석에 쓴 재료는 아크릴이다. 아크릴의 강성은 적합하지만 깨지기 쉬운 성질을 갖고 있기 때문에 질긴 성질을 가진 알루미늄을 이용했다.



<Stabilizer 서스펜션 모델링>

Stabilizer과 AGV 사이에 서스펜션을 달아 이중 서스펜션을 설치했다. 네 군데에 가이드라인과 서스펜션이 달려있는 모습이다.

나.2.3. 서스펜션 최적화

최적화 계산은 Matlab Global Optimization Toolbox를 사용하였으며, 알고리즘은 GA를 사용했다.

최적화 시 variables는 AGV 서스펜션, Stabilizer 서스펜션의 스프링 계수, 댐퍼 계수이다. Parameters는 캐스터 바퀴 및 모터 바퀴의 스프링 계수이다.

나.2.3-1 최적화 문제 정의

장애물 주행 중 발생하는 급격한 가속도의 변화는 유체 이송 안전성을 크게 저해한다. 급격한 가속도 변화를 최소화해야 한다고 판단하였다. 따라서 Stabilizer에 발생하는 가속도의 최댓값을 최소화하고자 했다.

스프링 계수와 감쇠 계수의 범위는 다음과 같다.

$$K \in <2, 40>(\text{N/mm}), C \in <100, 700>(\text{N}\cdot\text{m/s})$$

정의한 최적화 문제는 아래와 같다.

$$\begin{aligned} & \text{minimize} \quad |a_{\text{stabilizer}}|_{\max} \\ & \text{subject to} \quad K, C \end{aligned}$$

나.2.3-2 최적화 결과

최적화 계산 결과는 Table. 6과 같다.

$m = 8.5(\text{kg})$	$K_{sf}=4(\text{N/mm})$	$C_{sf}=0$
$M = 13.5(\text{kg})$	$K_s=8(\text{N/mm})$	$C_u=100(\text{N}\cdot\text{m/s})$
$m_f=0.1(\text{kg})$	$K_{sf}=16(\text{N/mm})$	$C_d=240(\text{N}\cdot\text{m/s})$
$\text{wheel} = 0.1(\text{kg})$	$K_{wf}=200(\text{N/mm})$	$C_{wf}=0$
	$K_{wr}=200(\text{N/mm})$	$C_{wr}=0$

Table. 6 최적화 계산을 통해 도출한 설계 변수 값.

나.2.4 실험

1m/s의 장애물 주행을 통해 서스펜션 실험을 진행했다. Fig. 9 장애물의 높이를 조

절하며 최적화한 서스펜션의 성능을 확인하였다. 성능의 판단기준은 유체 수면의 높이 변화, stabilizer plate의 z축 가속도이다. z축 가속도의 경우 IMU센서로 측정하였으며, 측정 위치는 plate의 정중앙이다. 매트랩을 통한 IMU센서 측정 방법과 매트랩 코드는 첨부목록 [B]-3, [B]-4 파일을 참고하면 된다. 이후 서스펜션의 최적화 전/후 비교, 서스펜션의 유무 비교를 진행하였다.

안정적인 유체 이송의 기준은 유체의 수면높이 변화량 20mm 이내로 설정했다.

감쇠 계수는 쇼크업소버 내부에 들어가는 실리콘 오일의 동점성 계수를 통해 계산[5]하였다. 이를 위해 쇼크업소버의 사이즈를 측정하였다. 또한 가정을 통해 단순화한 후 계산하였다. 가정은 아래와 같다. 또한 Engineering site를 참고하여 진행하였다. 사이트는 [B]-4에 첨부했으며 VPN을 사용해야한다.

1. Newtonian fluid
2. No-slip condition
3. Damper force, restore force는 동일
4. Damper force, restore force, spring force는 선형성을 갖는다.

감쇠 계수 계산은 정확하지 않다. 때문에 최적화 계산을 했더라도 스프링 계수와 동점성 계수를 계속 바꿔가며 실험을 진행했다.

나.2.4-1 서스펜션 성능 확인

a) 서스펜션 최적화 후

Fig. 9의 장애물 높이를 1mm 감소시키며 진행하였다. 그 결과 장애물의 높이가 4mm일 때 안정적으로 유체를 이송할 수 있었다. Fig. 11은 4mm 높이의 장애물 주행 중 유체 수면이 최대일 때를 나타낸다. 이 때 수면높이 변화량은 약 15mm이다. Fig. 12는 stabilizer plate의 z축 가속도를 측정한 것이다.

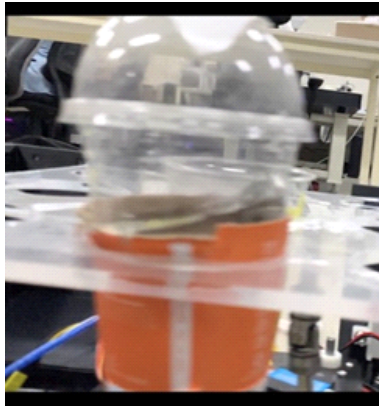


Fig. 11 최적화 후 유체 수면의 최대높이

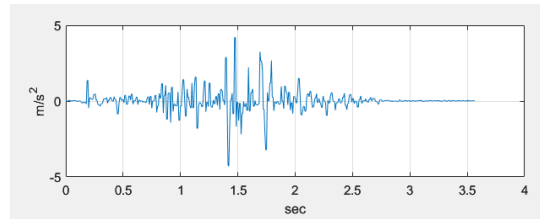


Fig. 12 최적화 후 stabilizer plate의 z축 가속도

b) 서스펜션 최적화 전

최적화 이전인 Table. 5 설계 변수로 설정 후 4mm 높이의 장애물 주행 실험을 진행했다.

Fig. 13은 유체 수면높이가 최대일 때를 촬영한 것이다. 이 때 수면높이 변화량은 약 27mm이다. Fig. 14는 stabilizer plate의 z축 가속도를 측정한 것이다.



Fig. 13 최적화 전 유체 수면의 최대 높이

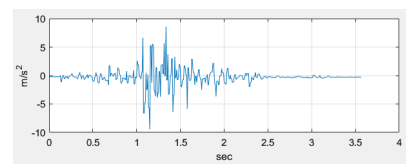


Fig. 14 최적화 전 stabilizer plate의 z축 가속도

c) 서스펜션이 없을 경우

서스펜션이 없을 경우 동일한 조건에서 실험을 진행했다. Fig. 15는 서스펜션이 없을 경우 유체의 모습을 보여주고 Fig. 16는 장애물 주행 중의 stabilizer plate의 z축 가속도이다.



Fig. 15 서스펜션X 유체의 모습

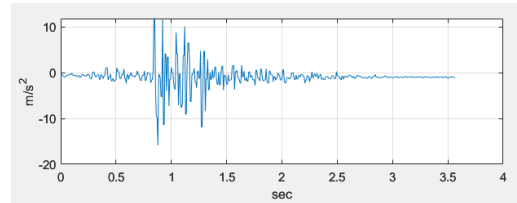


Fig. 16 서스펜션X stabilizer plate의 z축 가속도

d) 실험 데이터 정리 및 비교

실험 데이터를 Table. 7과 같이 정리하였다. 서스펜션을 최적화함으로써 최대 가속도 값과 RMS값이 감소했고, 유체 수면의 변화량도 감소하였다.

결과적으로 서스펜션을 사용함으로써 RMS값은 약 71%감소, 최대 가속도 값은 약 50%감소, 유체 수면 변화량은 약 63% 감소했음을 알 수 있다.

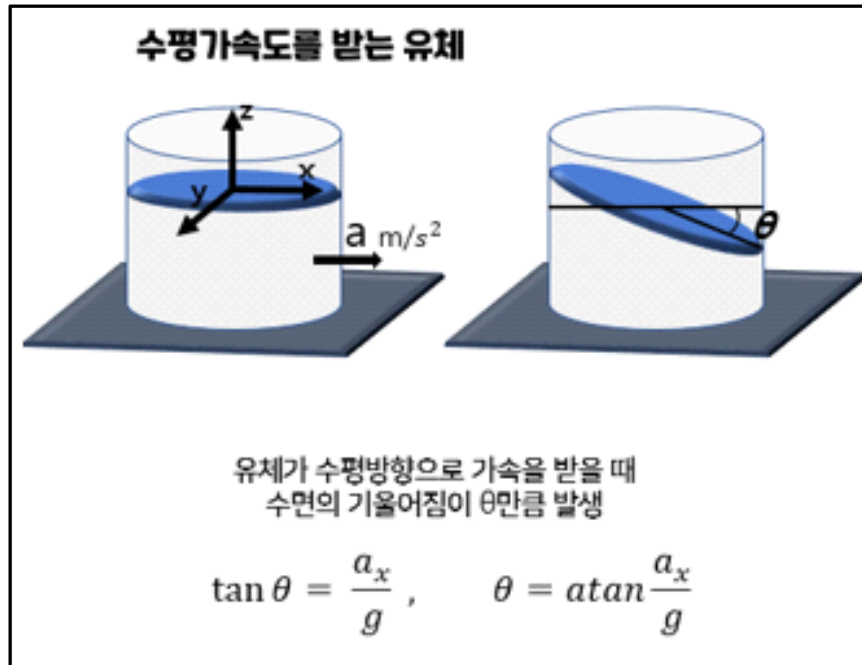
	RMS	최대 가 속도값	수면 변화량
최적화 후	0.7208	4.2709	약 15mm
최적화 전	1.4698	8.5941	약 27mm
서스펜션X	2.5259	15.878	약 41mm

Table. 7 실험 데이터

나.3. 하위제어(모터, 컨트롤러 등)

나.3.1. Stabilizer 제어

Stabilizer의 제어는 모터 제어와 플레이트 각도를 제어하여 가감속에 반응하는 유체 수면의 기울기를 제어하는 것을 말한다. 기본적으로 유체 수면의 기울기는 다음과 같은 식을 따라 변한다.



가속도를 받는 용기 안 유체의 정상상태 해석 및 각도

유체 용기를 이송하는 플레이트의 각도를 기울여 유체 수면의 기울기 변화로 인한 유체 넘침을 보정하고 안정성을 가지며 이송하는 것이 본 연구의 목표이다. 우리 작품의 시스템은 다음과 같은 구성요소를 가지고 있다.

- 1) 식음료를 올려 놓는 플레이트
- 2) 가속도에 대응하여 플레이트의 각도를 기울여주는 Stabilizer
- 3) Stabilizer를 운반하는 AGV

이중 플레이트와 AGV의 디자인은 Stabilizer 시스템과 디자인에 따라서 종속될 것이다. 공학적 문제를 어떻게 정의하고 어떤 것을 우선순위로 둘 것이냐에 따라서 AGV의 디자인과 플레이트의 디자인이 스테빌라이저의 디자인과 시스템을 결정할 수도 있을 것이다. 우리 작품의 Stabilizer 는 기본적으로 Z축 선형, Roll, Pitch 의

자유도를 가지는 3-DOF 시스템이다. 이러한 시스템이 제어적으로 가지는 장점은 다음과 같다.

- 1) Roll에 대한 자유도를 가지는 것으로 AGV의 회전, 방향 전환 시의 가감속에 대한 안정성을 확보할 수 있다.
- 2) 필요에 따라 Z축 선형운동을 통한 유사 서스펜션 시스템을 구축할 수 있다.
- 3) 2축 Gimbal 시스템을 구성할 수 있다.
- 4) 3개의 모터를 가지므로 토크에 대한 부담을 줄일 수 있다.

단점은 이와 같다.

- 1) 3개의 모터에 대한 제어의 부담이 존재한다.
- 2) 역기구학 풀이에 대한 제어 부담이 존재한다.
- 3) 복잡한 시스템으로 인해 유체 제어에 대한 변수가 늘어난다.
- 4) 3-DOF의 장점을 취하지 않으면 오히려 단점이 된다.

언급한 단점에 대한 부분은 이렇다. 3개의 모터에 대한 제어의 부담은 이렇하다. 기본적으로 아두이노와 같은 MCU의 PWM 단자는 최대한 적게 쓰는 것이 좋다. 레지스터를 조작하여 사용할 수 있는 출력부가 제한되어있을 뿐 아니라 엔코더를 사용한 모터를 사용할 경우 인터럽트를 입력할 수 있는 입력부의 단자 또한 제한되어 있다. 아두이노 우노의 경우 엔코더 DC 모터를 동시에 2개 밖에 사용할 수 없다. 인터럽트 단자가 4개 존재하기 때문에 인터럽트 단자가 더 많은 아두이노 메가의 사용을 강제한다. 역기구학 풀이를 아두이노에서 푸는 것 또한 적지 않은 부담을 준다. PID 제어의 알고리즘 등이 같이 들어간 코드를 동시에 아두이노 등에서 제어하는 것은 여러 부담을 준다. 이와 같은 제어적 부담이나 단점이 존재하나 3-DOF 시스템을 채택한 이유를 여러 PPT를 통해 알 수 있을 것이다. 그러나 제어적 입장에서 실용화 보다 성능을 우선시 한다면 시스템은 최대한 간단한 시스템으로 구성하기를 추천한다.

나.3.1.1 모터 선정 및 개발 과정

서보모터는 굉장히 간단한 제어방법을 가지고 있다. 서보모터의 제어 방법에 대하여는 첨부한 [\[아두이노를 통한 서보모터의 제어.PDF\]](#)를 참고. PWM 단자 하나를 통한 간단한 제어와, 5V의 낮은 전압에서 동작한다는 점, 피드백 제어를 내장하고

있기 때문에 엔코더를 통한 위치과악이 필요없이, 명령한 각도를 그대로 실행한다는 점에서 매력적인 모터이다. 그러나 사양보다 높은 토크가 지속적으로 가해질 시 내장된 피드백 제어 루프에서 조금씩 오류가 나거나 내부의 모터가 손상되어 백래시 등이 심해지는 문제가 발생하였다. 정확히는 높은 토크가 가해질 시 입력된 위치에서 조금씩 벗어나는 문제가 있었고, 이것이 모터를 손상시키므로 백래시가 심해졌다. 이러한 문제를 해결하기 위하여 엔코더 DC모터를 사용하여 3-DOF 시스템을 구성하였다. 그러나 DC모터의 자체 백래시가 존재하여 백래시 문제에서 벗어나기 어려웠다. DC 모터 선정 과정에서 유성기어 모터를 선정하였는데, 유성기어 모터는 감속비가 100:1 이 넘어갈 경우 효율이 떨어지기 때문에 감속비에 대한 토크 보상을 받지 못하므로 유의하길 바란다. 또한 엔코더 DC 모터는 절대 엔코더가 존재하지 않을 경우 초기 위치를 알지 못한다. 이것은 스텝모터도 같은 문제를 공유하고 있으나 DC 모터의 위치 PID 계인을 조정함에 있어서, 그 문제가 더욱 도드라지게 나타난다. DC 모터의 PID 제어가 완벽하게 되지 않을 경우 오버슈트로 인한 위치의 오차가 계속 누적되게 되므로, 절대 엔코더 혹은 다른 센서를 부착하여 위치에 대한 보정을 해주어야 한다. 추천하는 시스템은 스텝모터를 사용하여 각도를 정확하게 제어하고, 센서를 통한 위치 보정을 해주는 것을 추천한다.

토크에 대한 부분은 시스템이 요구하는 토크보다 2.5배 이상으로 할 것을 추천한다. DC 모터의 PID 제어는 [DC 모터 PID 속도 및 위치 제어.PDF]를 참고. PID 제어의 전반적인 부분은 [IMU와 스텝모터를 활용한 PID 응답특성.PDF]를 참고

나.3.2. AGV DC 모터 제어

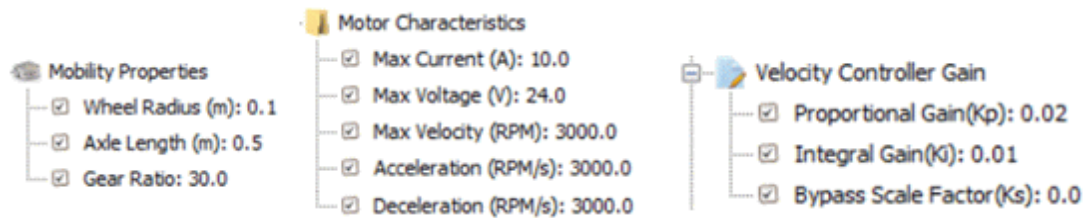
1)모터제어기 모터 특성 정립(Mobility Properties)

모터제어기를 통해 모터 2개에 명령을 주기 위해서는 회로 연결 후 PI 이득을 조정해야한다. 모터유틸리티 앱을 켜서 Scan Device를 하여 모터를 연결해준다.

그런다음 모터에 속도 명령을 넣어주는데, 이 속도명령을 잘 따라가는지 확인하기 위해서는 Real-time-Plot창을 열어 확인한다. command속도와 실제 모터속도의 차이를 그래프로 확인해가며 모터의 configuration을 확립해준다. 속도 제어기는 PI 제어기가 기본 구조이며 P와 I 이득을 조정한다. 조정의 최종 목표는 모터가 오버슈트나 진동 없이 빠르게 원하는 속도에 도달하는 것이다.

모터의 ch1, ch2 둘다 PI 값을 제대로 주었다면 모터는 명령대로 잘 움직일 것이다.

그리고 모터의 감속비(gear ratio)와 통신속도를 제대로 설정해 주어야 한다. gear_ratio 파라미터에는 이동로봇의 모터와 바퀴간 감속비율을 설정한다. 감속비율은 바퀴가 1회전 할 때 모터의 회전 비율을 의미한다.



2) 아두이노로 제어

-듀얼채널명령

아두이노 IDE 시리얼 명령으로 모터 2개를 제어하기 위해서는 모터제어기에 듀얼 채널 명령을 주어야 한다.

듀얼 채널 명령들은 로봇의 좌우 바퀴에 동시에 구동 명령을 내리고 상태를 읽어 오는데사용된다.

m_velocity_command, mvc	모터 채널 1,2의 속도 구동 명령을 내리고 (단위: RPM,RPM) 엔코더 값을 읽음(단위: pulse, pulse)
----------------------------	--

m_velocity_command는 좌우 모터에 속도 명령을 동시에 내린다. 그리고 좌우 모터의 위치를 읽어 온다. 좌우 모터를 500RPM, 500RPM 속도로 회전하기 위해서는 다음과 같이 입력한다. **mvc=500,500** 그러면 제어기는 좌우 모터의 현재 위치를 즉시 리턴하고, 명령을 실행하기 시작한다.

- RS-232 시리얼통신

아두이노에서 모터제어기로 명령을 주기위해서는 시리얼 통신을 사용해야한다.

PC의 아두이노 IDE 시리얼에 입력을 주고 그 값을 모터제어기에 전달하고 싶다면 Serial통신포트를 2개 사용해야한다. 하나는 PC에서 아두이노로, 하나는 아두이노에서 모터제어기로 주는 방식이다.

-속도프로파일사용

DC모터에 주는 속도 값은 RPM 단위로 들어가는데 우리가 사용할 속도 프로파일에서 나온 속도값은 m/s 단위이다. 따라서 속도 값을 RPM값으로 변환한후 명령을 주어야 한다.

나.3.3. IMU 센싱 및 필터링 구성

MPU-6050은 3 방향의 가속도 센서와 3 방향의 자이로 센서로 구성된 6축 센서이다. 6개의 데이터를 보정하면 현재의 자세를 추정할 수 있으며 Stabilizer를 통해 짐벌을 구현할 수 있다.

$$\rho = \arctan\left(\frac{A_X}{\sqrt{A_Y^2 + A_Z^2}}\right)$$

$$\phi = \arctan\left(\frac{A_Y}{\sqrt{A_X^2 + A_Z^2}}\right)$$

$$\theta = \arctan\left(\frac{\sqrt{A_X^2 + A_Y^2}}{A_Z}\right)$$

그림 2-37 가속도 데이터를 이용한 Roll, Pitch, Yaw 계산식

가속도 센서는 노이즈에 취약하고 자이로 센서는 장시간의 드리프트 현상을 방지하여야 하므로 Sampling Rate = 0.04를 통한 상보필터의 적용 만으로 현재의 자세를 추정하는데 충분하며, 16 MHz의 오실레이터를 가지고 있는 아트메가2560으로 도 센싱 및 필터링을 통해 자세를 측정하는 주기를 3ms 이하로 유지할 수 있다.

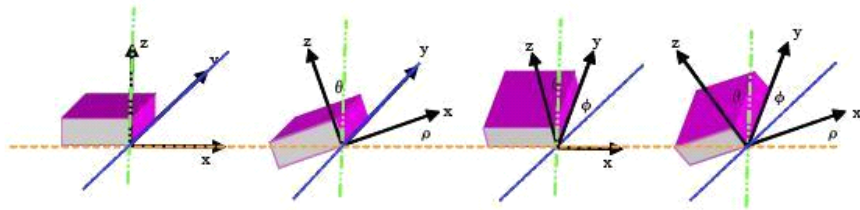


그림 2-38 IMU 센서를 이용한 자세 추정

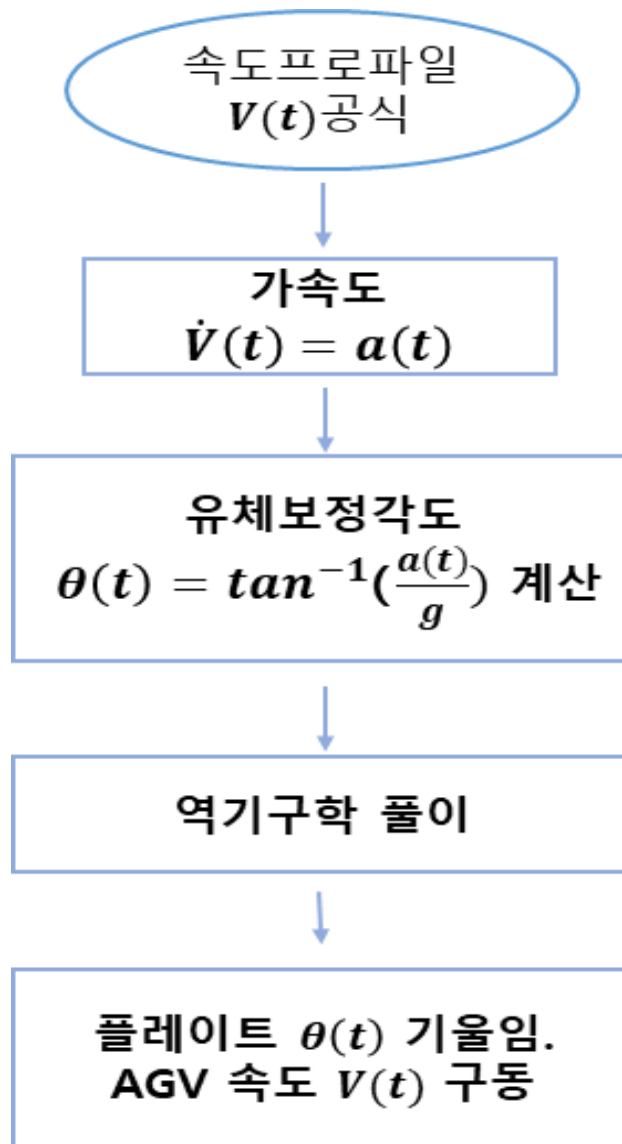
MPU6050의 계수를 조작하여 선형 가속도에 대한 반응성을 줄이는 방법에 대해서는 [IMU 상보필터 계수 조정 및 Stabilizer 성능 평가.PDF]를 참고. MPU6050은 6축 센서로 AGV의 글로벌 좌표 및 방향 제어 필요한 Yaw 데이터를 안정적으로 획득할 수 없으므로 지자기 센서를 포함한 9축 센서가 필요하다. 이에 대하여 다양한 기능을 가지고 있는 AHRS 센서인 이투스 사의 EBIMU IMU 센서를 사용하였다. 이에 대한 참고 자료는 [EBIMU] 폴더를 참고. 이 센서의 기능 중 하나는 중력 가속도를 제외한 선형가속도를 계산한다는 것이다. 이를 통해 Z축 선형 운동을 제어하여 유사

서스펜션을 구성할 수도 있고, 선형 가속도 데이터를 그대로 유체 정역학식에 넣어 프로파일 없이 클로즈드 루프로 이루어진 Stabilizer를 구성할 수도 있다. 이것은 2021년 캡스톤디자인 팀의 알고리즘 작동방식과 기본적으로 유사하다. 그러나 선형 가속도를 그대로 이용하는 것은 불안정하고 이것 또한 일종의 P제어이기 때문에 데이터에 대한 노이즈를 줄이고 안정된 데이터를 뽑아내는 것이 중요하다. 데이터 필터링에는 로우 패스 필터와 평균-이동 필터를 사용하였다. 로우 패스 필터를 코드 안에 구현하는 방법은 여러 가지가 있으나, 새로 들어온 데이터의 보정계수를 A, 기존 데이터의 보정계수를 B로 했을 때 $A + B = 1$ 이 되게 하고 새 데이터의 계수를 0.4 ~ 0.04에서 조절하는 일종의 더블 필터로 사용하는 것이 직관적이다. 데이터 보정을 둔감하게 할수록 반응성이 떨어져 오히려 유체 제어에서 낮은 성능을 보일 수 있으니 유의하길 바랍.

나.4. 협조 제어 알고리즘

Stabilizer가 제대로 작동하기 위해서는 가속도 데이터를 정밀하고 신속하게 측정하는 것이 중요하다. 그러나 가속도 데이터를 정밀하게 얻어내는 것은 쉬운 일이 아니다. 일상에서 자주 볼 수 있는 속도계나 거리계는 가속도 데이터를 이용하지 않고 바퀴의 회전 수를 측정하거나 GPS를 이용하여 계산한다. 정밀한 가속도 센서가 아니면 많은 노이즈가 생길 뿐 아니라 이러한 데이터를 필터링하고 적분하여 계산하기는 쉽지 않기 때문이다.

이러한 이유로 개념 설계 및 초기 과정에서 센싱을 통한 데이터를 얻기보다 DC모터의 회전수를 이용하여 가속도를 계산하였으며, 이를 바탕으로 속도 Profile 및 가속도 Profile을 구성하여 Engineering Sample을 실험 및 검증하였다.



AGV의 DC 모터에는 자기식 엔코더가 부착되어 있어 1회전 당 19 pulse를 감지할 수 있으며 감속비가 1:17이므로 64 (mm) 지름의 바퀴가 1 회전 할 때에 323번의 pulse를 감지할 수 있는 것이다. pulse 당 AGV가 이동하는 거리는 1.24 (mm)이며 pulse 및 시간을 이동거리-시간 의 함수로 나타내어 수치미분하면 현재 가속도 데이터를 얻어낼 수 있다. 이러한 계산은 현재 사용하는 문워커 모터 제어기 소프트웨어 안에서 자체적으로 일어나므로, 사용자는 원하는 DC 모터의 회전수를 입력하고 입력하는 회전수에 맞는 가속도 데이터를 계산하고 신뢰할 수 있다.

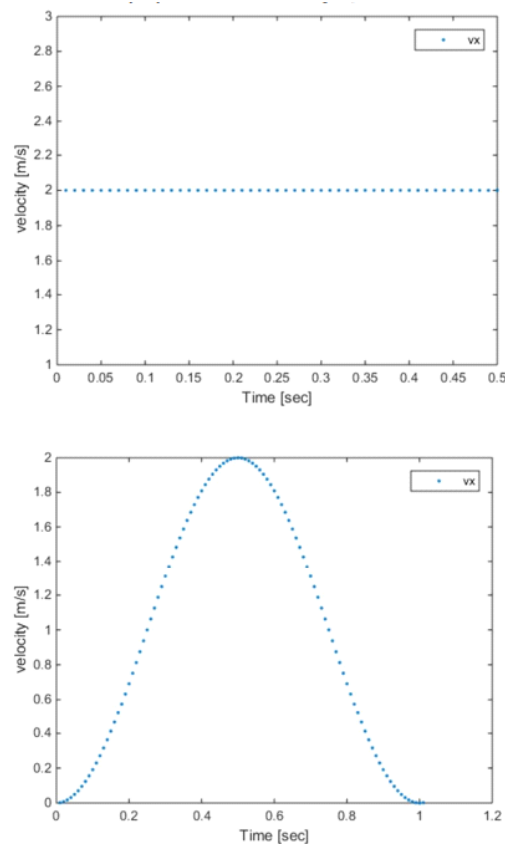
아두이노 알고리즘 안에서 가속도 데이터를 유체 정역학 식에 넣고 풀이하여 가속도 대응 플레이트 각도를 얻어내고, 역기구학 식을 통해 모터 1,2,3의 제어각을 얻어내어 명령하면 현재 가속도에 대응하는 각도로 플레이트가 꺾인다. 아두이노 안

에서 AGV와 Stabilizer 동시적으로 명령을 내리면 AGV의 실제 가속도와 플레이트 각을 안정적으로 제어할 수 있는 것이다.

나.4.1. 속도 프로파일 구성

속도 프로파일은 기본적으로 위치, 속도, 가속도와 시간과의 관계에서 얻을 수 있다. 본 논문에서는 플레이트의 최대각도와 전체 가감속 거리를 입력하면 최대속도와 최대가속도 그리고 걸리는시간을 출력하는 속도 프로파일을 구현했다.

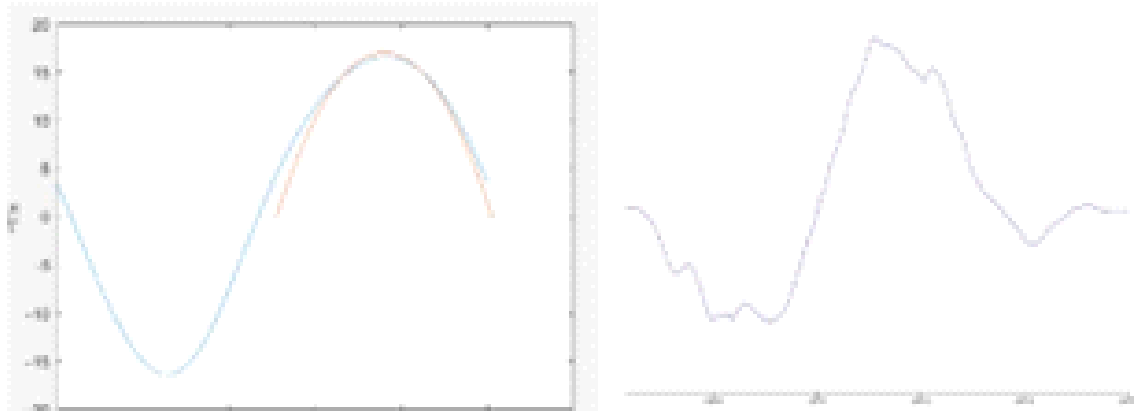
따라서, 같은 시간동안 Stabilizer의 플레이트 각도에 따라 AGV의 가속도를 일치화시킬 수 있고 유체를 효과적으로 제어한다. 먼저 플레이트 최대 각도를 매개변수로 입력하여 최대 가속도를 반환하기 위해서는 등가속도 운동 용기 속의 유체의 기울어지는 각도를 계산하는 방식을 이용해야한다. 본 논문에서는 플레이트의 목표 각도와 AGV의 목표거리만을 입력받아 최대가속도와 최대속도를 반환하는 속도 프로파일을 구현했다.



Trajectory Planning을 활용한
x-velocity profile

나.4.2. 유체 제어 알고리즘 구성 및 실험

유체 제어 알고리즘은 AGV와 속도 프로파일을 미분한 가속도 프로파일과 Stabilizer의 유체 보정식의 가속도 프로파일을 일치시키는 Open-loop 제어를 구성하는 것으로 하였다. AGV의 바퀴, DC 모터의 RPM 프로파일을 미분하여, 가속도 프로파일로 만든 뒤, 이것을 유체 보정식에 넣어 나온 보정각을 그대로 Stabilizer에 대입하는 방식이다. 중요한 것은 AGV의 속도 프로파일과 실제 속도 프로파일이 동일한지, 가속도 프로파일과 실제 가속도가 일치하는지이다. 유체의 수면 기울기는 가속도와 연관이 있으므로 실제 가속도를 EBIMU의 선형가속도 측정기능을 통하여 측정하였다. 이 그래프는 AGV가 아닌 테스트베드 환경에서 실험한 그래프이다.

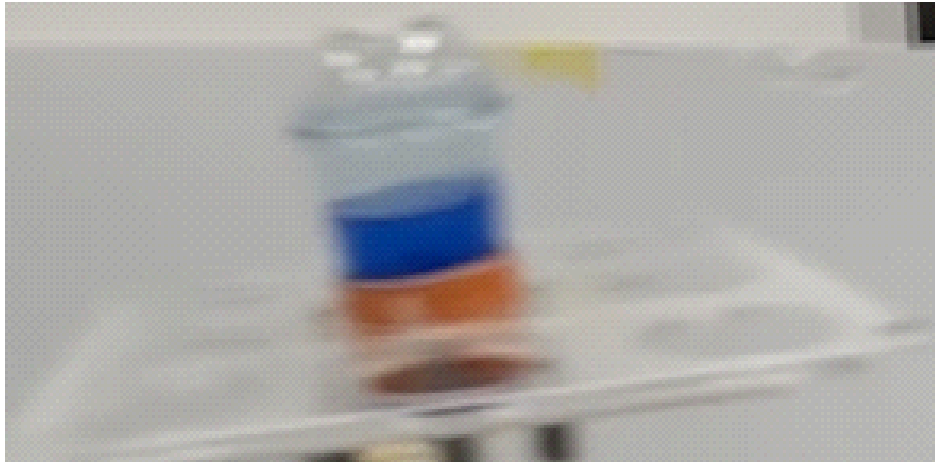


가속도 프로파일과 실제 측정한 가속도 그래프

이러한 가속도 그래프를 통해서 여러 가지 변수로 인해 노이즈가 발생하지만 대략적으로 일치하는 모습을 볼 수 있다. 그러나 가속도 프로파일이 대략적으로 일치하는 것으로 실제 유체 보정에 대한 성능을 크게 기대할 수 없으므로, 이러한 노이즈에 대한 해결과제가 남아있다.



가속 시 유체 보정 능력 확인(5장의 이미지를 겹침)



가속 시 유체 보정 능력 확인(5장의 이미지를 겹침)

실제 주행 환경에서의 다양한 변수로 인한 노이즈가 생기나, 그래프가 대략적으로 일치하는 것을 볼 수 있다. 이러한 주행환경에서 유체 보정의 효과를 측정하기 위해 측정된 가속도 그래프와 프로파일이 거의 일치하는 꼭짓점 부근에서의 유체 거동을 촬영하였다.

가속도 2.5 m/s^2 의 주행환경에서 반복 촬영한 사진의 불투명도를 낮추어 겹쳐 비교해본 결과 같은 부근에서의 유체 거동이 매우 흡사하고, Stabilizer의 기울기와 유체 수면의 기울기의 차이가 평균 3° 이하로 나타났다. 유체 정역학식에 의하여 Stabilizer의 기울기가 15° 로 기울어질 때, 이러한 각도와 평행한 수면의 움직임을 보인다. 이러한 결과로 인하여 가속도 프로파일과 실제 주행 시 가속도가 일치할 때 Stabilizer를 통한 유체 제어의 효과가 있음을 알 수 있다.

그러나 일치하지 않은 부근에 대한 해결책 마련이 필요하고 이러한 부분을 Closed-Loop 제어 방식을 통해 해결하기를 권장한다. EBIMU의 선형가속도 계산을 통하여 선형가속도 데이터를 확보하고, 필터링하여 보정하는 방법으로 반응성에 대한 성능을 확보할 수 있을 것이다. 그러나 필터링 계수를 과하게 설정하면 오히려 반응성이 떨어져 성능이 낮아지거나, 모터의 토크 및 RPM을 잘 따져서 반응성에 대한 불리함이 없도록 시스템을 설정하는 것이 중요할 것이다.

나.4.3. ROS를 활용한 상위통합 명령제어

속도 프로파일은 기본적으로 위치, 속도, 가속도와 시간과의 관계에서 얻을 수 있다. 본 논문에서는 플레이트의 최대각도와 전체 가감속 거리를 입력하면 최대속도와 최대가속도 그리고 걸리는시간을 출력하는 속도 프로파일을 구현했다.

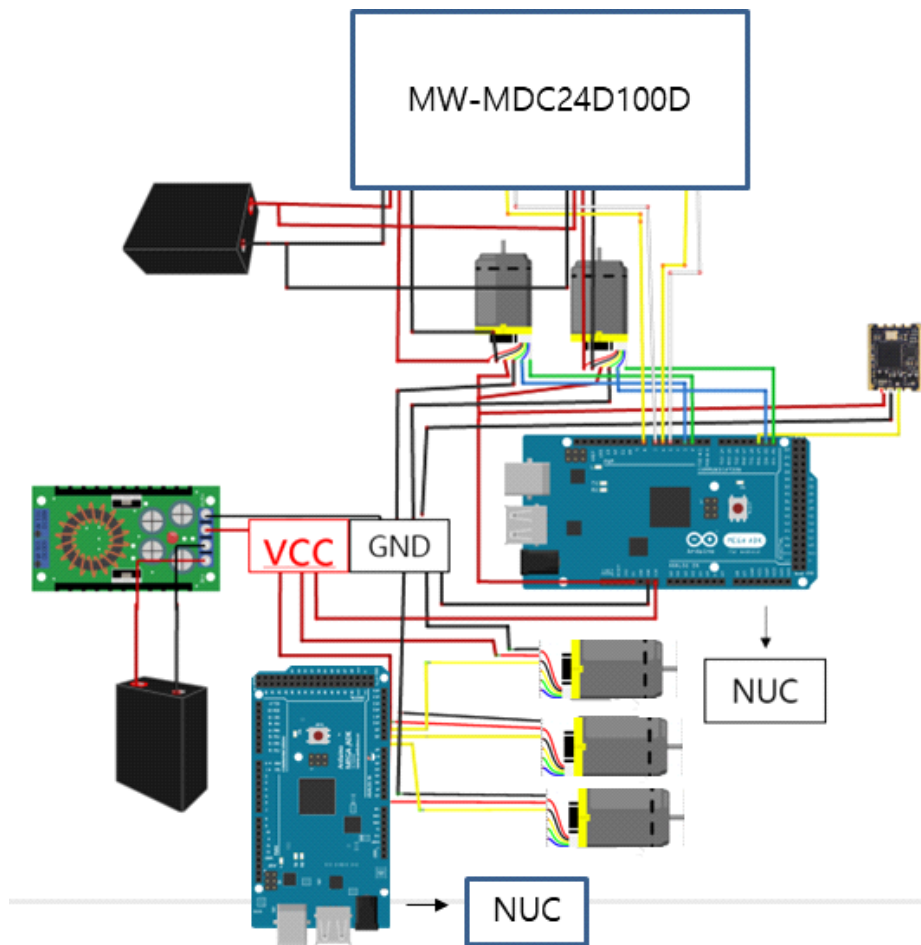
따라서, 같은 시간동안 Stabilizer의 플레이트 각도에 따라 AGV의 가속도를 일치화 시킬수 있고 유체를 효과적으로 제어한다. 먼저 플레이트 최대 각도를 매개변수로 입력하여 최대 가속도를 반환하기 위해서는 등가속도 운동 용기 속의 유체의 기울 어지는 각도를 계산하는 방식을 이용해야한다. 본 논문에서는 플레이트의 목표 각도와 AGV의 목표거리만을 입력받아 최대가속도와 최대속도 를 반환하는 속도 프로



파일을 구현했다.

나.5 제어부 및 회로설계

나.5.1 AGV 및 Stabilizer 회로도



AGV와 Stabilizer MUC로 각각 아두이노 메가를 사용했다.

나.5.2. 사용 부품

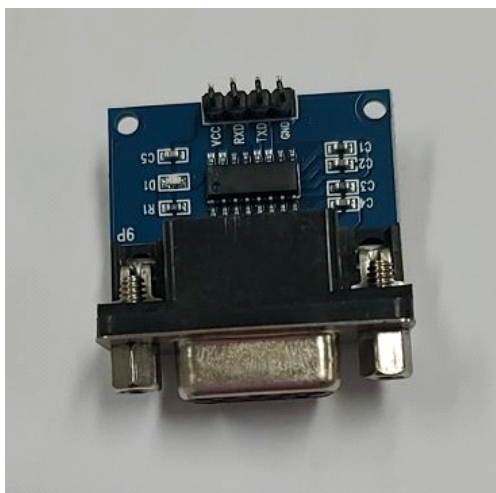
②모터제어기: Moon Walker 모터제어기

1개 혹은 2개의 DC 모터를 좀 더 강력하게 구동하기 위해 이 제품을 선택했다. MoonWalker 시리즈는 기본적으로 전류/속도/위치 제어기를 탑재하고 있어서 로봇 암이나 주행로봇, 산업용 기기 등을 연구부터 실제 응용까지 아주 광범위한 적용이 가능하다. Motor Control UI 유틸리티를 사용하여 PC에서 손쉽게 제어기의 각 종 설정을 조절할 수 있으며 제어기의 상태 확인을 그래프로도 확인할 수 있다. 특히 직관적이고 손쉬운 다양한 구동 방식을 제공하므로 본 제품의 구매 후 간단히 제품의 연결에 관련된 내용을 숙지하고, 일반적인 시리얼 통신 포트나 혹은 Mini-USB 단자를 이용하여 PC와 연결한 후 MotorControl UI 유틸리티를 통해 내용을 확인하면 된다.

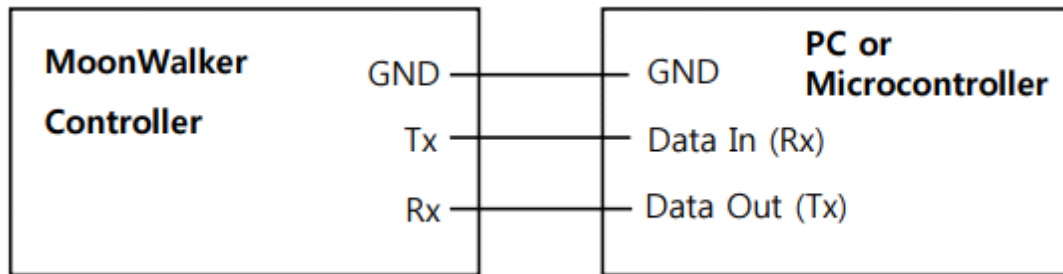


③통신모듈: MAX3232 RS232 to TTL 컨버터 모듈

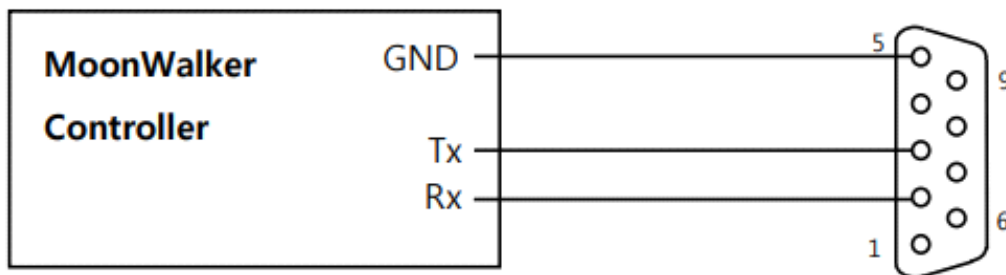
모터 제어기는 RS-232 시리얼 통신을 통해서 PC에 접속할 수있다. 이 연결은 제어기에 명령을 전송하고 제어기에서 다양한 상태 정보를 읽어오는데 사용된다. USB연결에 비해 노이즈에 강하고 선의 길이 제한이나 통신 오류 발생시 복구 가능성이 높기 때문에, 제어기가 실제 현장에 배치되는 경우 RS-232연결을 권장한다.



RS-232연결시 핀 번호를 제대로 확인하여 연결해주어야 한다. PC의 RS-232 포트는 보통 DB9 수(male) 커넥터로 구성된다. 그래서 제어기로부터 연결되는 신호 선들은 DB9 암(female) 커넥터에 결선 된다. 다음 그림을 참조하여 제어기의 Rx, Tx, GND

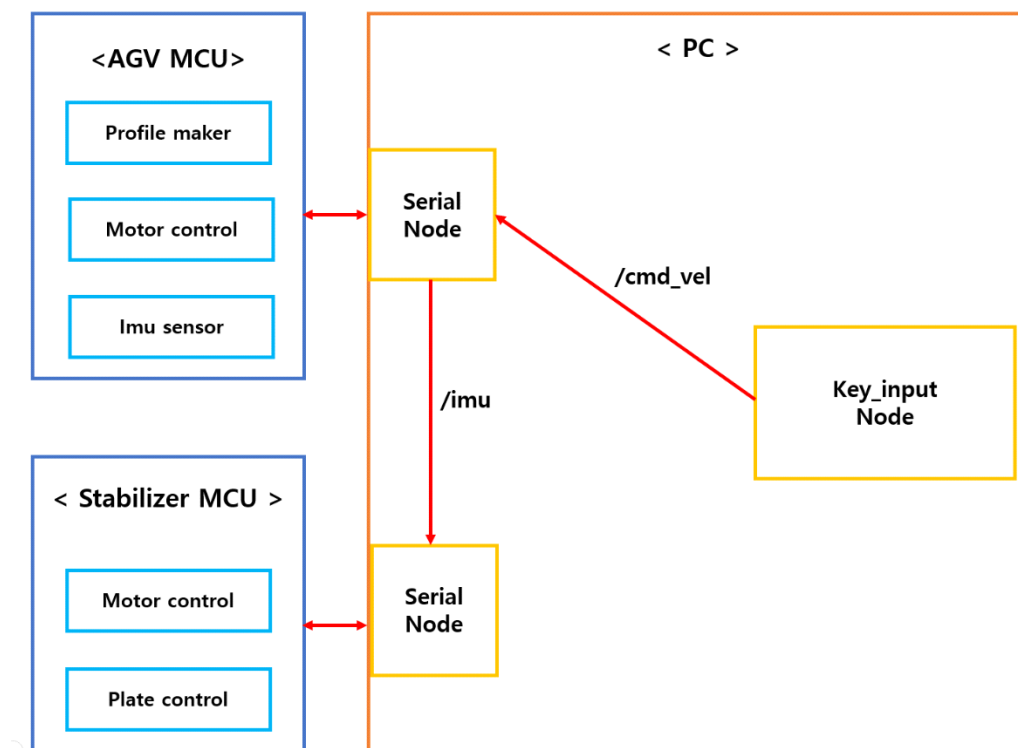


핀을 DB9 암 커넥터의 핀으로 연결한다.



나.6. ROS를 활용한 상위통합 명령제어(김시현)

인텔누크 소형PC 에 우분투 18.04를 설치하고 ros melodic버전으로 ROS제어를 했다.



상위통합제어로는 ROS를 활용했다. AGV와 Stabilizer 각각의 아두이노에

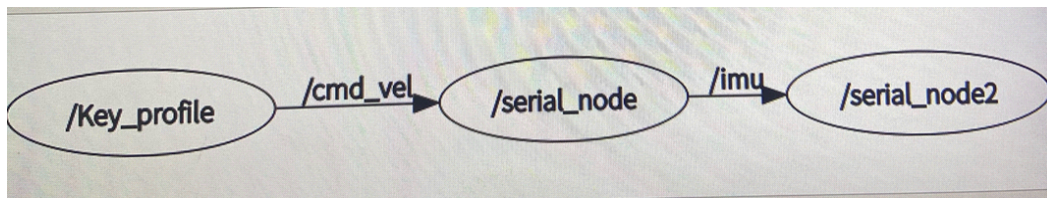
Rosserial을 통해 명령을 하고 두 개가 토픽 값을 주고 받기 위해서는 ROS활용해야 편리했다.

AGV 아두이노에 먼저 명령을 내리는 방법으로는 키보드 Key입력을 사용했다.

위 그림은 ROS 통합제어알고리즘이고, 대한 상세 설명은 다음과 같다. python 언어로 Key_profile이름 인 python 언어로 publisher 노드를 만들어 key 입력을 받으면 roserial 통신방법으로 serial_node를 통해 AGV아두이노에 /cmd_vel토픽에 값을 전달하도록 했다. AGV아두이노에서 /cmd_vel 토픽으로부터 명령을 받으면 AGV는 속도프로파일과 imu센서를 통해 값을 계산하여 AGV DC모터를 프로파일대로 구동시키고, 또 다른 roserial 통신인 serial_node2를 통해 /imu토픽에 imu 데이터와 가속도 프로파일 값을 보내 Stabilizer의 DC모터를 구동시킨다.

전체적으로 serial_node는 두 개가 사용됐으므로, 하나는 serial_node, 하나는 serial_node2라는 이름으로 구분시켜 주었다.

토픽값이 주고받아지는 형태를 rqt_graph를 통해 보면 다음과 같다.



프로젝트 시 사용한 연구실 모바일로봇 인텔 누크에 설치한 catkin_ws 작업환경 속 패키지의 이름은 'practice2' 이고 여기안에 launch 파일과 node파일이 있다. 또한 아두이노 코드도 모두 있다. (첨부 3,4,5)

라. 여름방학 그룹스터디

4. 김시현 그룹스터디 자료

4.1 아두이노 DC모터 제어관련 (첨부1)

Index

- I. Arduino소개
- II. 기본회로
- III. 통신
- IV. 모터/모터제어기
- V. 아두이노로 제어

아두이노와 문워커 모터제어기 간 RS232통신을 활용해 DC모터의 RPM을 제어하는 방법을 첨부하였다. 문워커 모터제어기 내에서 모터의 configuration을 정립하는 방법과 , 모터의 PI제어하는 방법, 기본 회로도도 설명되어있다.

4.2 ROS설치 및 기본 활용 개념 관련(첨부2)



ROS 활용을 위해 우분투 18.04버전 설치부터 ros melodic 버전 설치에 도움이 된 사이트들을 첨부하였다. 또한 토픽개념들과 rosterial통신에 대한 내용을 담았다.

A. 참고 문헌

[1] Li J, Wang J, Chou W, et al. Inverse kinematics and dynamics of the 3-RRS parallel platform. In: IEEE international conference on robotics and

automation, Seoul, Korea, 21-26 May 2001, pp.2506-2511.

[2] Optimization Toolbox for use with MATLAB. User guide, Version 2. The Mathworks Inc, Natick(USA, 2004)

[3] Ana Luísa Marques de Carvalho,” Improvement Project of an AGV's Suspension System in order to Reduce Manufacturing Costs” , Instituto Superior Técnico, p. 2, November, 2019

[4] B. Bounab, Multi-objective optimal design based kineto-elastostatic performance for the DELTA parallel mechanism, Robotica 34 (2016) 258-273

[5] A Syrakos, Theoretical study of the flow in a fluid damper containing high viscosity silicone oil: Effects of shear-thinning and viscoelasticity, Physics of Fluids(2018)

B. 참고 자료



220712study자료_
양성백.pptx

1.



suspension_matla
b.m

2.



imu.m

3.



220822study자료_
양성백.pptx

4.

5. <https://www.tribology-abc.com/calculators/damper.htm>



8월23일 발표_이
인호 (1).pptx

6.



배터리포항어셈
블리333 - 복사
본.iam

7.

8. 이동건 Stabilizer 제어에 대한 부분은 이동건 자료를 참고.

[EBIMU] 폴더

[DC 모터 PID 속도 및 위치 제어.PDF]

[IMU 상보필터 계수 조정 및 Stabilizer 성능 평가.PDF]

[IMU와 스텝모터를 활용한 PID 응답특성.PDF]

[아두이노를 통한 서보모터 제어.PDF]