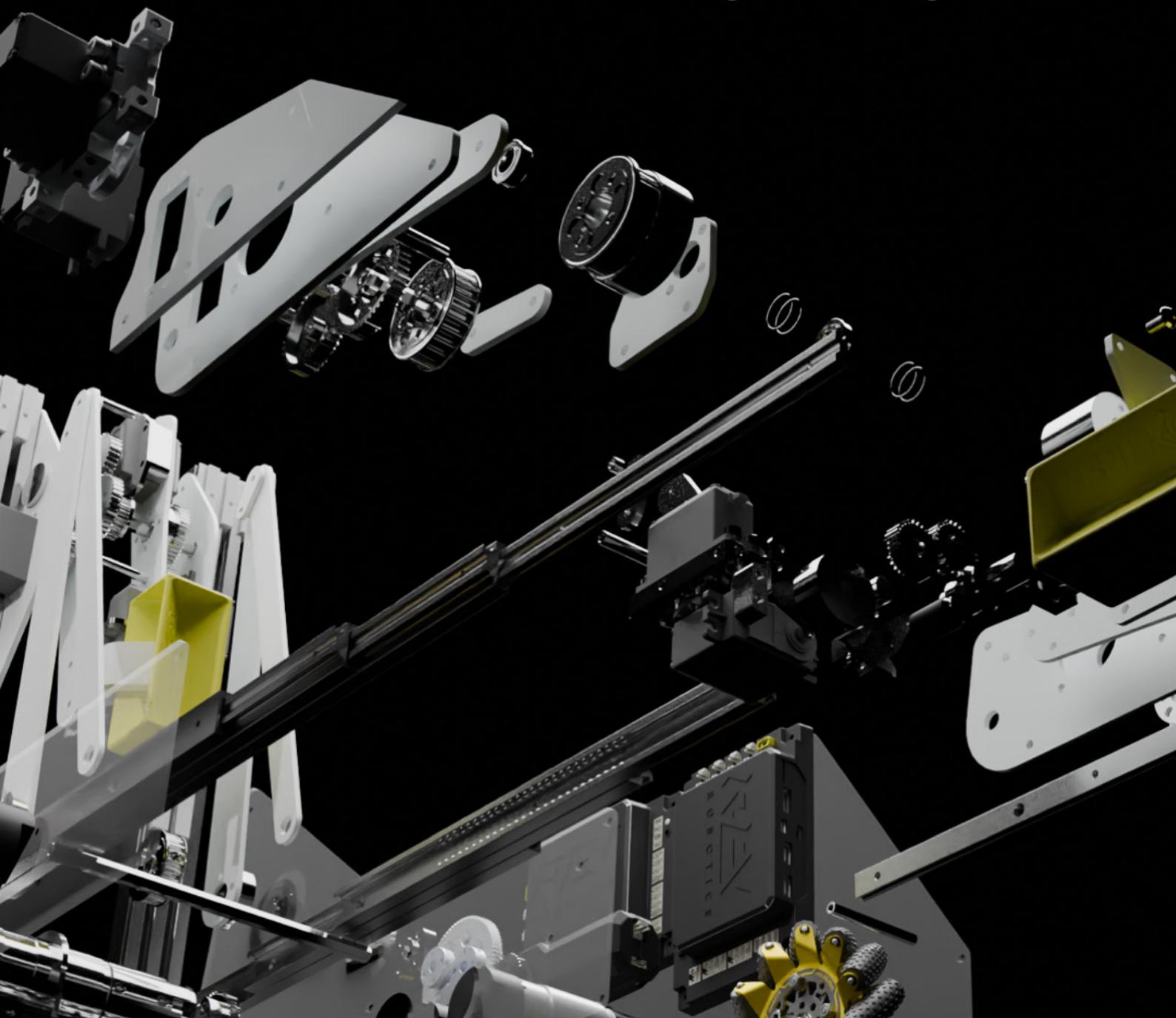


#25309

TALOS

2024-25

Engineering Note



KRC TALOS 엔지니어링 노트

편집 | 23 김준성 (xsorexia@gmail.com)

시즌 | 2024-25 FIRST DIVE : Into the DEEP

날짜 | 2024/12/17 - 2025/01/15

Team TALOS는 로봇공학 연구회 KROS의 연구회원들로 구성되며,
본 연구회는 한국과학영재학교와 과학기술정보통신부의 지원 하에 운영되고 있습니다.

TEAM
MANAGEMENT
팀 운영 및 관리

해당 섹션은 TALOS 팀의 예산과 인원 선발,
작업 절차 등에 대해 다룹니다.

목차 |

1. 팀 및 인원 소개
2. 예산 계획 및 관리

I. 팀 및 인원 소개

1. Team TALOS

1) 팀 소개

TALOS (#25309)는 FIRST Tech Challenge (FTC) 팀입니다. 로봇공학 연구회 KROS의 연구회원 14명으로 구성된 TALOS는 2018-19 Rover Ruckus 시즌부터 코리아로봇챔피언십 (KRC)에 참여해 왔습니다. 작년 2024-25 CENTERSTAGE 시즌에는 Inspire Award를 수상하는 쾌거를 이루었으며, 한국 대표팀의 자격으로 휴스턴에서 개최된 FTC 세계대회에 참여하였습니다.



TALOS 팀명의 유래

고대 그리스 신화에서 대장장이 헤파이스토스는 크레타를 방어하고자 한 제우스의 요청으로 최초의 로봇 탈로스를 만들었다고 전해집니다. 우리의 팀 이름은 헤파이스토스가 이성을 통해 창조한 탈로스처럼 멋진 로봇을 만들어내겠다는 의지를 표방합니다.



TALOS의 공식 색상은 노란색, 검은색, 흰색이며 상징은 투구의 모양을 하고 있습니다. 기존에는 한국 팀 번호 #5073을 사용하였으며, 작년 시즌부터 #25309를 사용하고 있습니다. 해당 로고를 활용한 단체 티셔츠와 배지를 제작하였습니다.

2) 팀 연혁

TALOS는 2018-19 시즌부터 꾸준히 KRC에 참여해왔으며, 24-25 Into the DEEP이 합산 6번째 시즌입니다. 졸업한 선배님들께서 멘토로 도움을 주시며 발전을 이어나가고 있습니다.





2018-19 ROVER RUCKUS

🏆 N / A



2019-20 SKYSTONE



2021-22 FREIGHT FRENZY

🏆 Think Award (1st EN)



2022-23 POWERPLAY

🏆 Connect Award (2nd EN)



2023-24 CENTERSTAGE

🏆 Inspire Award



2024-25 INTO THE DEEP

**2023-24 FIRST World Championships**

CENTERSTAGE 시즌에는 KRC에서 Inspire Award를 수상하여 미국 휴스턴에서 개최된 FIRST Championships에 한국 대표로 참가하였습니다.

총 10번의 경기 중 4번 승리하였으며, 해외 팀들과 교류하며 로봇 설계 및 제작에 적용할 수 있는 다양한 기술과 작업 방식을 배울 수 있었습니다. 세계 대회에서의 경험은 24-25 시즌 로봇을 설계함에 있어 큰 도움이 되었습니다.



2. 인원 소개

1) 빌더 - Builder

빌더는 로봇을 설계하고 제작하는 과정을 담당합니다. 로봇의 다양한 부분에 어떠한 메커니즘을 적용할지를 테스트하고 결정하며, 이를 바탕으로 로봇을 제작합니다. 또한, 로봇의 구조에 따라 달라지는 게임 전략을 프로그래머들과 함께 의논합니다.



23 홍정우 | Jungwoo Hong
Team Captain / Builder

23학번 팀 캡틴으로써 24-25 Into the Deep 시즌을 이끌고 있으며, 팀의 운영을 맡고 있습니다. 빌더로써 로봇의 제작에 참여합니다.



22 장호원 | Howon Chang
Builder

23-24 CENTERSTAGE 시즌 World Championship에서 캡틴으로써 팀을 이끌었으며, 빌더로써 로봇의 제작에 참여합니다.



22 김수기 | Sugi Kim
Builder

22학번 빌더로써 로봇의 제작에 참여하며, 이번 시즌 로봇의 전반적 설계를 담당하였습니다.



22 최성빈 | Sungbin Choi
Builder

22학번 빌더로써 로봇의 제작에 참여합니다.



22 이규진 | Gyujin Rieh
Builder

22학번 빌더로써 로봇의 제작에 참여합니다.



22 강현빈 | Hyeonbeen Kang
Builder

22학번 빌더로써 로봇의 제작에 참여합니다.



22 정의진 | Uijin Jung
Builder

22학번 빌더로써 로봇의 제작에 참여합니다.



23 김진용 | Jinyong Kim
Builder

23학번 빌더로써 로봇의 제작에 참여합니다.

**23 류승완** | Seungwan Ryu

Builder

23학번 빌더로써 로봇의 제작에 참여합니다.

**24 이소민** | Somin Lee

Team Captain / Builder

빌더로써 로봇의 제작에 참여하며, 24학번 팀 캡틴으로 선정되었습니다.

**24 함주원** | Juwon Ham

Builder

24학번 빌더로써 로봇의 제작에 참여합니다.

2) 프로그래머 - Programmer

프로그래머는 로봇을 구동하는 코드를 작성합니다. 로봇을 구상, 설계하는 과정에 함께 참여하며, 로봇을 제작하는 과정에서도 각 부분을 테스트할 수 있도록 코드를 제공합니다. 완성된 로봇이 TeleOp, AutoOp에 모두 맞추어 작동할 수 있도록 Onbot Java를 사용합니다.

**22 유태우** | Taewoo Yu

Team Captain / Programmer

23-24 CENTERSTAGE 시즌 KRC 캡틴으로 팀을 이끌었으며, 22학번 프로그래머로써 코딩에 참여합니다.

**23 김준성** | Joonsung Kim

Programmer / Outreach

23학번 프로그래머로써 코딩에 참여하며, 해외 팀 교류 및 웹사이트 관리, 엔지니어링 노트 정리 등 대외 활동을 담당합니다.

**24 이재빈** | Jaebin Yi

Programmer

24학번 프로그래머로써 코딩에 참여합니다.

3) 코치 / 멘토 - Coach / Mentor



김호숙 선생님 | Dr. Hosook Kim

Coach

한국과학영재학교 수리정보과학부 학
부장을 맡고 계시며, TALOS 팀의 지
도를 맡고 계십니다.



19 이승찬 | Seungchan Lee

Mentor / Alumni

한국과학영재학교 19학번이며 이전
시즌에 팀 캡틴으로 참여하셨던 멘토
로써 팀을 지원해 주십니다.

II. 팀 운영 및 예산 관리

1. 팀 운영

1) 후원 (스폰서)

TALOS는 과학기술정보통신부의 지원을 받아 운영되고 있습니다. 학교를 통해 운영 예산을 제공받으며, 이를 사용해 로봇 제작에 필요한 부품과 공구, 부속품을 구입하여 사용하고 있습니다. Dream Design Center라는 메이커 공간을 지원받아 작업합니다.



2) 팀 운영 방식 (Workflow)

TALOS는 앞서 인원 소개에서 구분한 바와 같이 빌더와 프로그래머, 아웃리치 세 가지로 업무를 분담하며, 각각 11명, 3명, 1명의 학생들이 담당하고 있습니다. 아래의 그림에서 볼 수 있듯이, 기간에 따라 진행하는 작업이 정해져 있으며 앞선 작업이 마무리되어야 다음 작업이 원활하게 진행될 수 있습니다. 또한, 로봇의 설계 및 구상 작업, 아이디어 회의 등에는 업무 구분 상관없이 모든 학생들이 참여함으로써 보다 넓은 관점에서 생각할 수 있다는 장점이 있습니다.

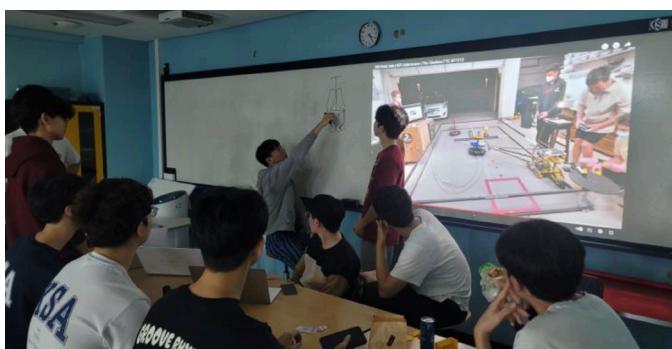
학생들의 거주 지역이 전국에 분포하여 있으며 기숙사 학교의 특성상 방학 동안 함께 작업을 진행하기가 어렵습니다. 따라서 학기 중에 여러 차례 회의를 진행하고, 경기 2주 전부터는 학교에서 잔류하며 로봇을 제작합니다. 아래 사진은 학기 중 회의 사진입니다.



△ 09월 08일 | 미션 공개 시청



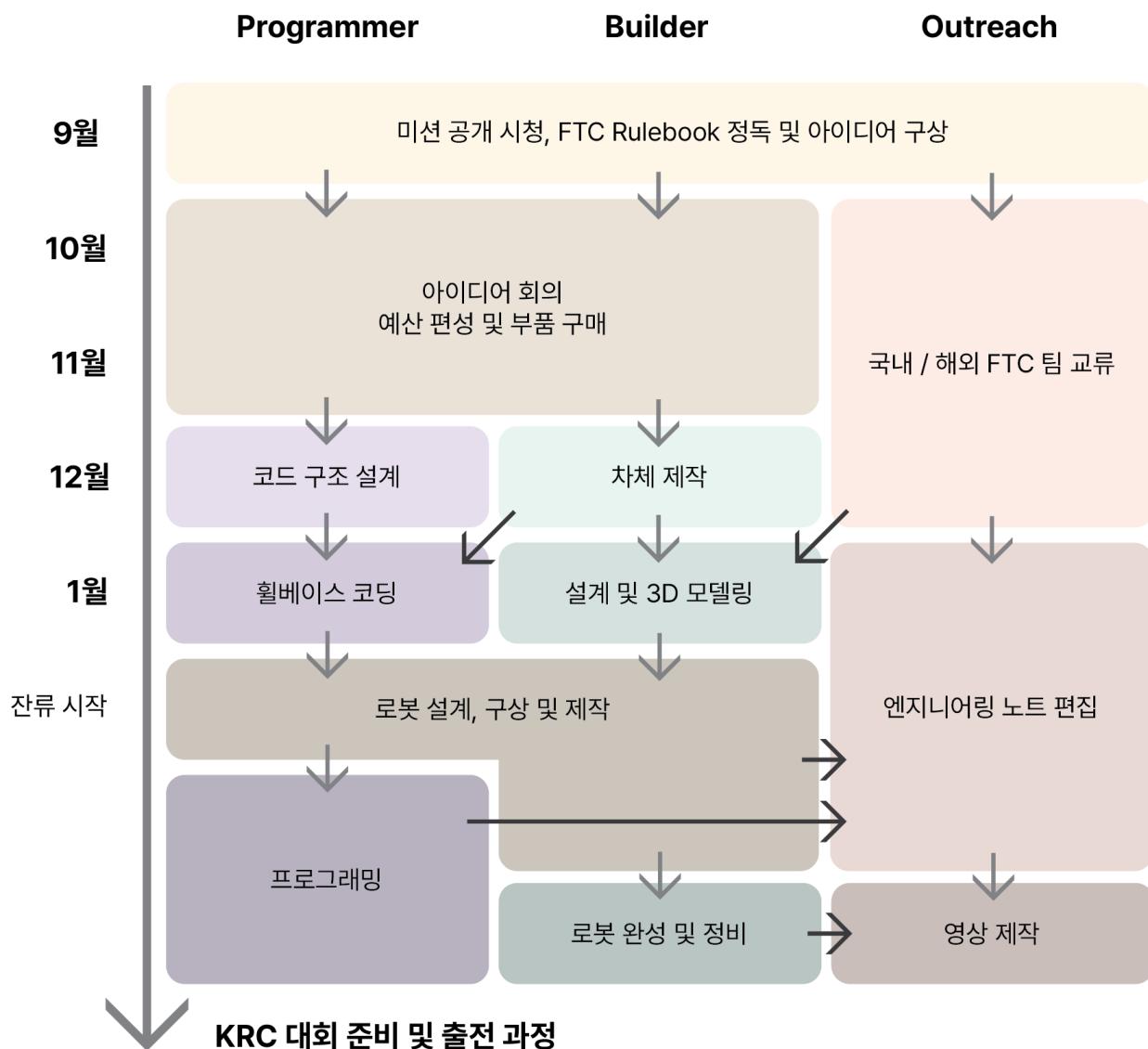
△ 09월 09일 | 김호숙 선생님과 미션 관련 회의



△ 09월 30일 | Intake 방식 회의



△ 01월 12일 | 잔류 시작 후 회의



아웃리치 (대외활동) 담당은 국내/외 FTC 팀들과의 교류를 담당하며 엔지니어링 노트 편집, 영상 제작 등을 함께 진행합니다. 해외 팀들과는 인스타그램을 통해 소통하고 미팅 일정을 정해 Zoom, Google Meet 등으로 회의를 진행합니다. 해외 팀들과 미팅 시 일부 빌더와 프로그래머가 함께 참여하며, 아웃리치 담당이 미팅을 진행합니다.

엔지니어링 노트는 Google Docs에 팀원 모두가 각각 정리하고, 이를 하나의 문서로 취합하여 Adobe InDesign으로 편집합니다. 하나의 업무를 같이 진행한 사람들 중 한 명이 담당을 맡아 엔지니어링 노트 섹션의 총괄을 맡아 사진 등을 추가합니다.

2. 예산 관리

1) Into the DEEP 경기장

AndyMark에서 FIRST Tech Challenge INTO THE DEEP Partial Game Set을 구입함. 경기장 기물의 1/2를 배송받아 충분히 테스트를 할 수 있으리라 판단함.

2024년 09월 11일에 구입하였음. 비용은 \$390.00 이었으며, 배송비로 \$274.42 를 지출하였음.



사용한 비용

\$664.42

2024년 09월 11일
AndyMark에서 구입

2) Misumi 리니어 슬라이드

해외 팀 영상을 참고하고 미팅 등에서 물어본 결과 많은 팀들이 미스미 슬라이드를 많이 사용한다는 사실을 알게 됨. 커스텀하기 쉬우며 가벼워 사용하기 편하겠다고 판단하였으며, SAR230, SAR240을 구입함.

2024년 09월 23일에 구입하였음. 비용은 ₩192,376 을 지출하였음.



사용한 비용

₩192,376

2024년 09월 23일
Misumi Korea에서 구입

3) GoBilda 부품 주문

GoBilda에서 다양한 부품을 주문함.

2024년 11월 26일, 12월 1일, 2025년 01월 06일에 각각 \$585.69, \$600.96, \$313.75 를 구입했으며 매번 배송비로 \$50 를 사용함.



사용한 비용

\$1,650.40

3번 분리 주문
GoBilda에서 구입

4) Axon 서보

Misumi 리니어 슬라이드와 마찬가지로 해외 팀들을 참고했을 때 Axon 서보를 사용한다는 사실을 알게 됨. 출력이 훨씬 강하여 안정적으로 사용할 수 있어 6개를 주문함.

2025년 01월 06일에 주문했으며, 물품 \$479.94 와 배송비 \$39.99 를 사용함.



사용한 비용

\$519.93

2025년 01월 06일 주문
Axon Robotics에서 구입

총 사용 금액 및 계획

약 총 430만원을 사용하였습니다. 모든 예산 지원은 한국과학영재학교로부터 지원받아 사용하였기에, 다음 시즌에는 다른 곳에서 스폰서를 받아보고자 합니다. 또한, 올해 많은 금액을 지원받아 다음 시즌에는 이번 시즌 구입한 물품들로 운영할 수 있을 것이라고 생각합니다.

OUTREACH 대외 활동

해당 섹션은 TALOS 팀이 국내/외 팀들과 교류하는 등 다양한 대외 활동의 과정과 결과물을 담고 있습니다.

목차 |

1. 과학축전 부스 운영
2. 웹사이트 제작 및 자료 공유
3. 외부 FTC 팀 교류

I. 과학축전 (KSASF) 부스 운영

1. KSASF 준비

1) KSASF란?

KSASF는 Korea Science Academy Science Fair의 약자로, 한국과학영재학교에서 매년 여름 진행하는 과학축전입니다. 홀수 년도에는 해외의 영재교육 기관들을 대상으로 한 국제 행사로, 짝수 년도에는 초등학생과 중학생들을 대상으로 한 국내 행사로 운영됩니다. 2024년도에는 국내 행사로 개최되었으며, 140명의 학생들이 참가하였습니다.

저희는 로봇을 사용하여 미션을 수행하는 부스를 운영하는 것이 수, 과학에 관심을 가지는 어린 학생들에게 좋은 경험이 될 것이라고 생각하였습니다. 로봇공학과 FTC 대회를 어린 학생들에게 알릴 수 있는 기회라고 생각해 여름방학 동안 부스를 준비하고 운영하였습니다.



2) 부스 준비 과정

2024년 7월 29일부터 8월 5일까지 학교에 잔류하며 부스를 준비했습니다. 빌더 5명, 프로그래머 3명이 작업에 참여하였으며, 미션을 직접 설계하고 로봇을 제작하여 학생들이 직접 조종하며 점수로 경쟁할 수 있는 부스를 운영하였습니다.



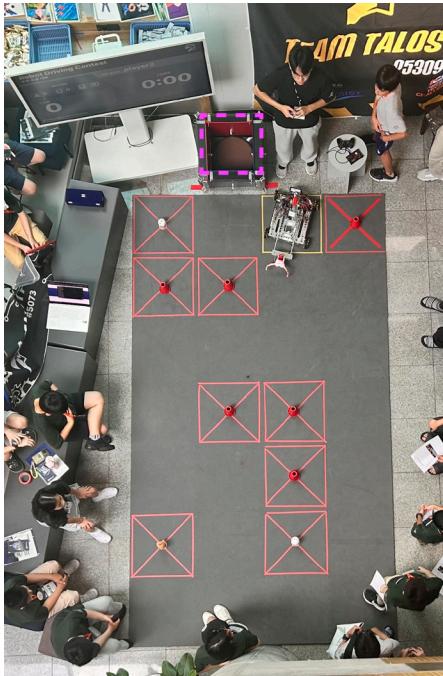
빌더 | 22 장호원, 22 강현빈, 24 이소민, 24 이시우, 24 함주원

프로그래머 | 22 유태우, 23 김준성, 24 이재빈

24학번이 대회 전 로봇 제작 경험을 쌓을 수 있는 좋은 기회였으며, 엔지니어링 노트도 별도로 작성하며 로봇 제작 과정을 미리 경험할 수 있었습니다. 작성한 엔지니어링 노트는 별도로 첨부하였습니다.

2. 부스 운영

1) 미션



KSA-RC (Korea Science Academy Robotics Competition)는 로봇공학 연구회 KROS에서 진행하는 팝업 부스입니다.



행사 타임라인

KSA-RC에 참가해보세요!
행사 타임라인은 아래와 같으니 참고하세요.

1. 행사 등록하기

직접 경기를 즐기고
다른 친구들과 경쟁하세요!



2. 규칙 익히기

직접 경기를 즐기고
다른 친구들과 경쟁하세요!



3. 경기 즐기기 / 경쟁하기

직접 경기를 즐기고
다른 친구들과 경쟁하세요!

왼쪽 위의 사진과 같이 노란색 박스에서 로봇이 출발합니다. 빨간색 격자를 제외한 곳으로만 움직일 수 있으며, 빨간색 격자 내에는 집을 수 있는 오브젝트가 있습니다. 오브젝트를 집어 바스켓 (분홍색)에 넣을 수 있으며, 오브젝트의 종류에 따라 점수가 달라집니다. 제한시간은 1분으로 진행하였습니다.

오브젝트는 빨간 콘 6개, 공 2개, 그리고 인형 1개를 두었습니다. 오른쪽 위의 설명을 출력하여 부스 앞에 부착하였으며, 학생들이 대기하며 컨트롤러 조종법을 익힐 수 있도록 설명해주었습니다.

또한, 아래 사진과 같이 경기를 진행함에 따라 점수를 셀 수 있는 프로그램을 직접 제작하여 (22 유태우) 학생들이 남은 시간과 현재 점수를 확인할 수 있도록 하였습니다. 부스 운영 전 한국과학영재학교 학생들을 대상으로 게임을 진행해 보고, 피드백을 받았습니다.



2) 운영

8월 6일 교내 1층에서 부스를 운영하였으며, 참가를 신청한 60명의 학생 중 37명이 실제로 참여하였습니다. 많은 학생들이 관심을 보였으며, 어린 친구들의 질문들에 답해주는 시간을 가질 수 있었습니다.



해당 행사를 브이로그 형태로 직접 촬영 및 편집하여 유튜브에 게시하였습니다. ([YouTube 링크](#))



만난 학생들

60 명



직접 참여한 학생들

37 명

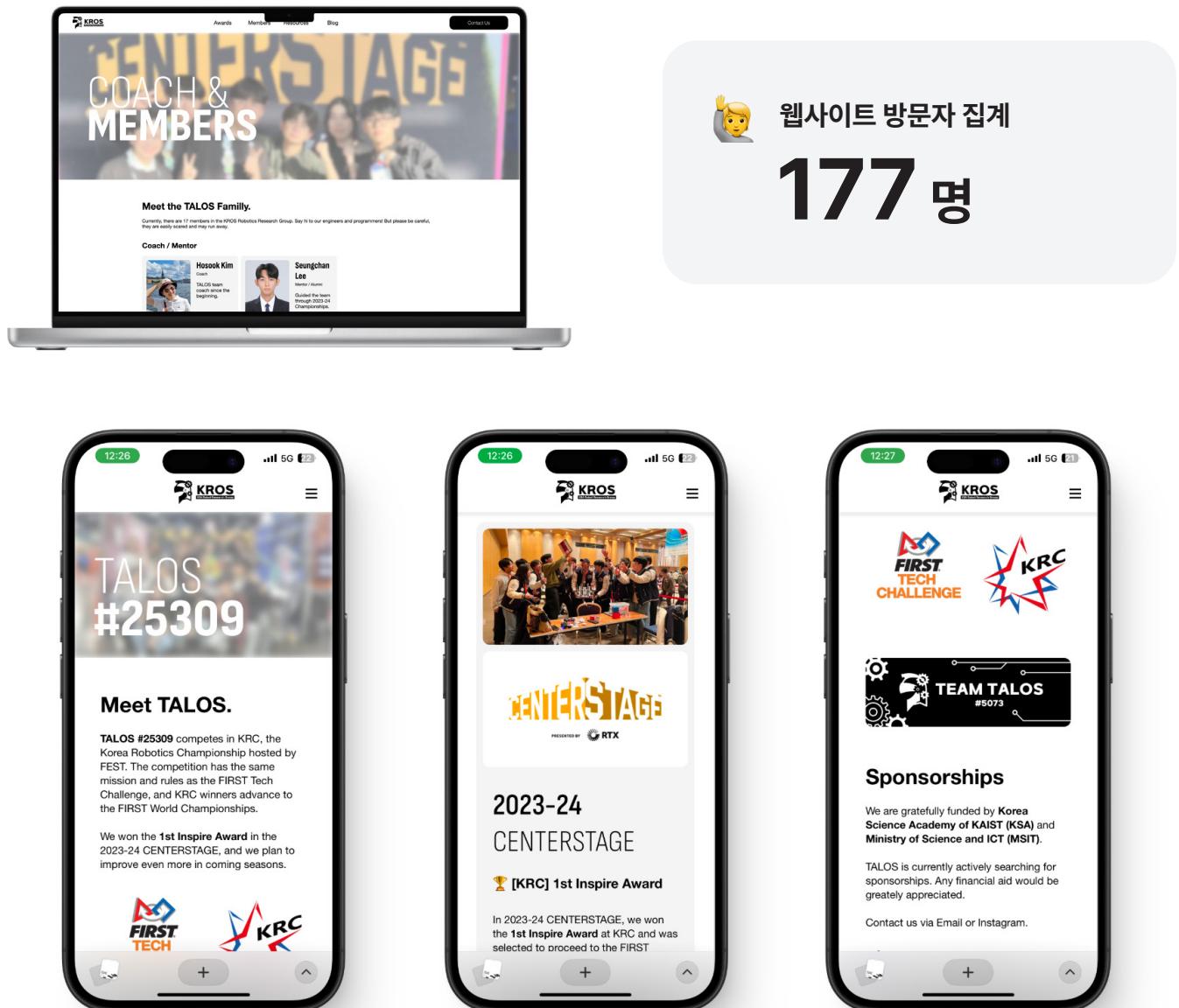
II. 웹사이트 제작 및 자료 공유

1. 웹사이트 제작

1) TALOS 소개 웹사이트

Team TALOS를 소개하는 웹사이트를 제작하였습니다. 해당 웹사이트를 통해 Team TALOS를 소개하는 동시에 엔지니어링 노트와 포스터, 세계대회 가이드 등 다양한 자료를 공유하고자 하였습니다.

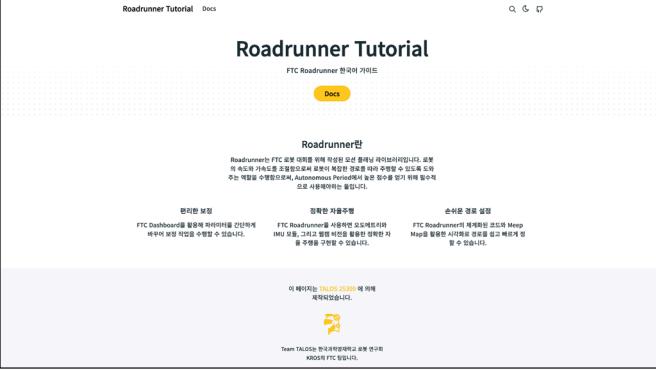
talosftc.com 도메인을 연결하여 사용하고 있으며, 매주 50명 정도가 웹사이트를 방문하는 것으로 집계되었습니다. 매년 새로운 내용을 추가하고 자료를 업로드할 예정입니다.

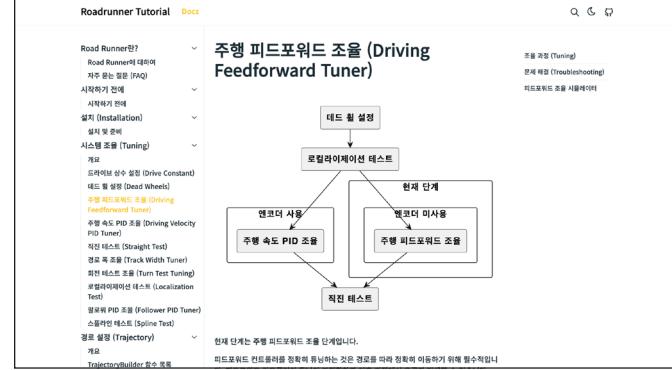


2) RoadRunner 가이드 웹사이트

Road Runner는 자율 주행시에 오도메트리와 IMU 센서를 활용한 자율 주행 기능을 편리하게 제공하기 위해 ACME Robotics 등이 제작한 모션 플래닝 라이브러리입니다. Autonomous Period에서 많은 점수를 얻는 것이 전체적인 순위에서 굉장히 중요한 FTC 특성상 요즘 대다수의 강력한 팀들은 Road Runner를 사용하여 정확한 자율 주행을 수행합니다. 하지만 국내에서는 다양한 이유로 인하여 아직 Road Runner를 적극적으로 활용하여 높은 자율 주행 성능을 구현하는 팀이 별로 없습니다.

우리는 한국 FTC 팀들이 Road Runner를 좀 더 쉽게 접하고, 적극적으로 사용할 수 있도록 Road Runner 사용법을 한국어로 정리한 웹사이트(guide.talosftc.com)를 제작하였습니다. 이 웹사이트는 초보자도 쉽게 Road Runner를 사용할 수 있도록 쉽고 직관적으로 작성되었으며, 시뮬레이션이나 상수 생성기 등 다양한 기능을 제공하기에 많은 한국 FTC 팀들에게 도움이 될 것이라 전망합니다.





2. 자료 제작 및 공유

1) FIRST Tech Challenge 세계대회 가이드북 제작

Team TALOS에서는 2023-24 시즌 FIRST Tech Challenge Championship을 준비하며 비행기표 예매, 호텔 예약, 일정 조정 등에 대한 경험이 부족해 많은 시행착오를 겪었습니다. 세계대회 경험이 처음이었으며, 참고하거나 물어볼 팀을 찾지 못하였습니다. 대회 이후, 한국에서 세계대회에 참여하게 될 팀들 역시 비슷한 문제들을 겪으리라 예상하여 출전 준비를 도울 수 있도록 가이드북을 제작하였습니다.



The cover of the 'FIRST Tech Challenge World Championship Guidebook' for Team TALOS #25309, dated 2024. It features the 'FIRST' logo in white and red, and the text 'World Championship 가이드북' and 'TALOS 2024' at the bottom.

항공편	출발	도착
KE1106 (대한항공)	PUS 부산 2024년 04월 16일 (화) 07:00	IOM 인천 2024년 04월 16일 (화) 08:10
KE031 (대한항공)	ICN 인천 2024년 04월 16일 (화) 09:20	DPW 일리노이스 2024년 04월 16일 (화) 08:00
AA3472 (아에로인천항공)	DPW 일리노이스 2024년 04월 16일 (화) 10:28	HOU 휴스턴 2024년 04월 16일 (화) 11:50
AA2941 (아에로인천항공)	JFK 뉴욕 2024년 04월 16일 (화) 08:10	ORD 시카고 2024년 04월 17일 (수) 10:52
KE038 (대한항공)	ORD 시카고 2024년 04월 21일 (일) 12:20	ICN 인천 2024년 04월 22일 (월) 16:50
KE1419 (대한항공)	ICN 인천 2024년 04월 22일 (월) 18:25	PUS 부산 2024년 04월 22일 (월) 19:35

Tip 1. 대회 시작일보다 일찍 도착할 수 있도록 예매하세요!

- 대회 전날 Team Registration 이 진행됩니다. 이때 팀 등록을 하고 명찰 등을 받아야 대회장 입장이 가능하니 꼭 참고해주세요.
- 혹시라도 비행기를 놓칠 수 있으니 어느 정도의 여유를 두고 예약해주세요. (최대 중요)

Tip 2. 항공편 간 시간 차를 충분히 두고 예매하세요!

- 경유편의 특성 상 한 비행기를 놓치거나 연착, 결영되는 경우가 발생할 시 다음 비행기를 놓치게 될 수 있습니다. 저희는 첫 비행기가 연착되어 댈러스에서 휴스턴으로 가는 비행기를 놓쳐 항공사에서 잡아 준 다른 비행기를 타고 시간에 겨우 맞추어 도착했습니다.
- 최소 5시간 정도의 여유를 두고 항공편을 잡는 것을 추천드립니다.

2) FIRST Dashboard 및 계정

대회 참가자들은 모두 FIRST 계정을 만들어 해당 팀 소속으로 등록이 되어야 합니다.

- 링크: <https://my.firstinspires.org/Dashboard>

해당 자료는 talosftc.com 웹사이트에 공유하였습니다. 크게 다른 내용은 다음과 같습니다.

1. 준비 및 대회장 이동

- 1) 비행기표 예매
- 2) FIRST Dashboard 및 계정
- 3) 숙소 예약
- 4) 로봇 운송
- 5) 배지 및 현수막, 단체티 제작
- 6) 엔지니어링 노트, 포트폴리오, 포스터
- 7) Promote video
- 8) 기타 준비 및 주의사항

2. FTC 대회 안내

- 0) 대회 관련
- 1) Team Registration / Unloading
- 2) Inspection
- 3) Judging
- 4) Match / Alliance

III. 외부 FTC 팀 교류

1. 해외 FTC 팀 교류

2024 FIRST World Championships에 참여하며 해외 팀들은 화상 미팅 등 다양한 플랫폼을 통해 서로 교류하는 것을 볼 수 있었습니다. TALOS에서도 해외 팀들에게 로봇 설계 및 제작 과정 등에 대해 질문하고 우리의 작업 방식을 보여주며 많은 것을 배울 수 있을 것 같다고 생각하였습니다.

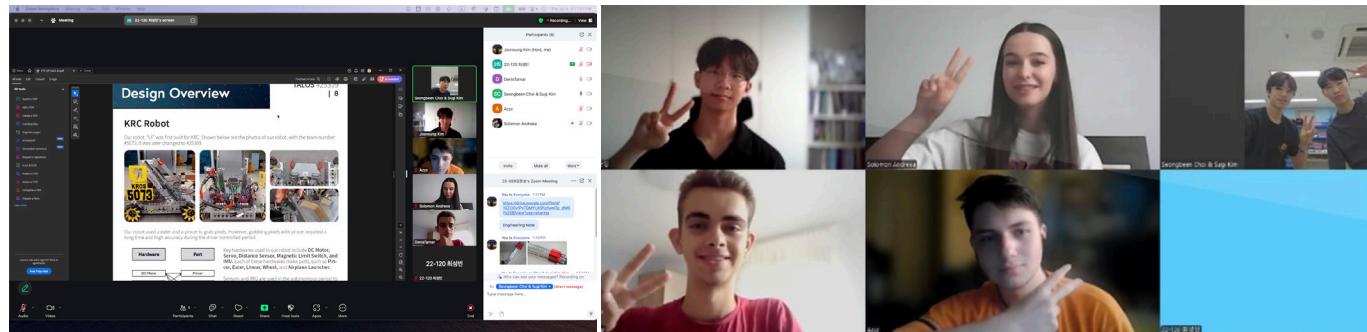
Team TALOS 역시 2024년 여름부터 2025년 겨울까지 다양한 국가의 해외 팀들과 교류하는 시간을 가질 수 있었습니다. 다양한 국가에서의 FTC 대회와 준비 과정, 팀의 운영 방식, 로봇 디자인과 프로그래밍, 대회에서의 경험 등을 교류함으로써 KRC를 준비하는데 도움을 받고, 우리의 경험을 공유하였습니다. Instagram DM 및 이메일을 통해 해외 팀들에게 연락을 먼저 하거나 미팅 요청을 받아 진행되었습니다.

1) 22114 Teoretika (로마니아 FTC팀)



날짜 | 2024년 07월 04일

참여 인원 | Teoretika 3명, TALOS 3명



Teoretika는 올해가 두 번째 시즌인 팀으로, 인스타그램을 통해 Zoom 미팅을 가지자고 제안하여 7월 4일 미팅을 진행함. TALOS에서 22 김수기, 22 최성빈, 23 김준성이 참여함.

각 팀의 Engineering Note / Portfolio를 공유하며 2023-24 CENTERSTAGE 시즌의 로봇 설계와 디자인, 프로그래밍과 관련하여 내용을 교류함.

로봇의 제작과 관련된 부분 뿐만 아니라, 팀원 선발 과정 및 설계 과정 등도 교류함. Teoretika에서는 로봇 제작에 참여하고 싶다는 학생을 제한 없이 모두 참여시키되, 과정을 지켜보며 이중 일부를 선발하여 대회에 참여 할 수 있도록 한다고 함 (대회 인원 제한 15인).

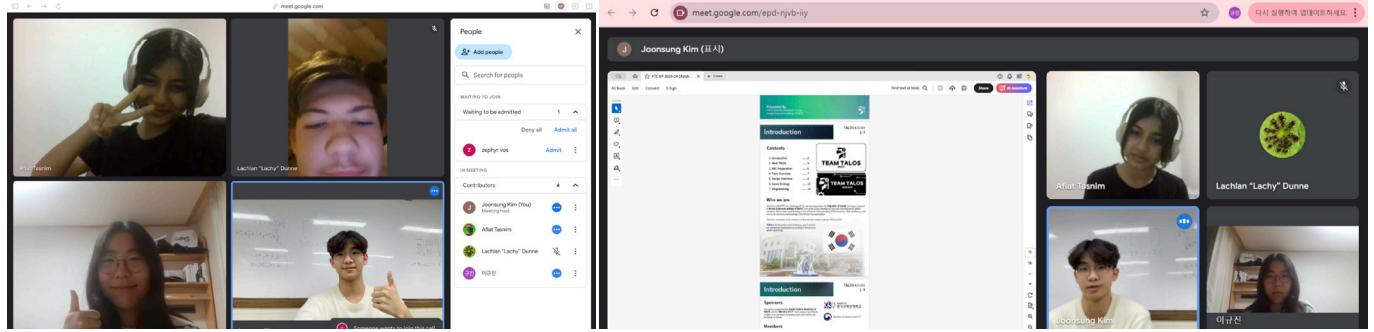
해당 팀에서는 CENTERSTAGE 시즌 웹캠을 사용하여 AprilTag와 Randomization Object를 인식하는 방식을 채택하였기에, 이를 구현한 방식과 코드에 대해 질문하고 자료를 공유받음.

로마니아와 한국의 FTC 시즌과 대회 진행 방식에 대해 설명하고 공유함. TALOS에서는 올해 세계대회에서의 경험을 공유함.

2) 18439 RoboKings Thorium (호주 FTC팀)

날짜 | 2024년 10월 25일

참여 인원 | RoboKings Thorium 2명, TALOS 2명



2024 FTC 세계대회에서 만난 12993 RoboKings Aurum 팀의 학생을 통해 18439 RoboKings Thorium 팀과 Zoom 미팅 일정을 잡음. TALOS 팀에서 22 이규진, 23 김준성이 참여함.

RoboKings Thorium (이하 RoboKings) 팀의 대부분은 올해가 첫 FTC 시즌으로, TALOS 팀에서 저번 시즌을 어떻게 준비했는지와 이번 시즌 준비를 어떻게 하고 있는지 질문함. 작년 세계대회에 제출한 Engineering Portfolio를 공유하며 TALOS의 로봇 디자인 및 설계 과정을 공유함. 또한, 이번 시즌 로봇 디자인에서 high basket에 집중하는 전략이 필요할 것이라는 논의를 함.

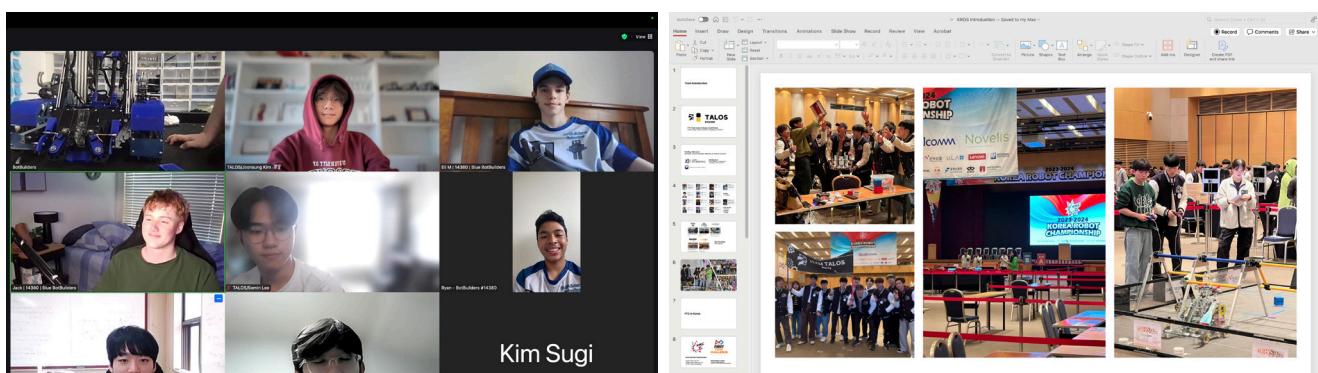
FTC에 참여하며 개인적으로 배우고 느낀 점 등을 공유했으며, 한국의 시즌 일정 및 준비 과정 등을 설명함.

RoboKings는 11월 2일에 첫 Regional competition이 있어, 지역 예선 이후 대회 전략 등에 대해서 교류하고 의논하기로 함.

3) 14380 Blue BotBuilders (호주 FTC팀)

날짜 | 2025년 1월 4일

참여 인원 | Blue BotBuilders 4명, TALOS 5명



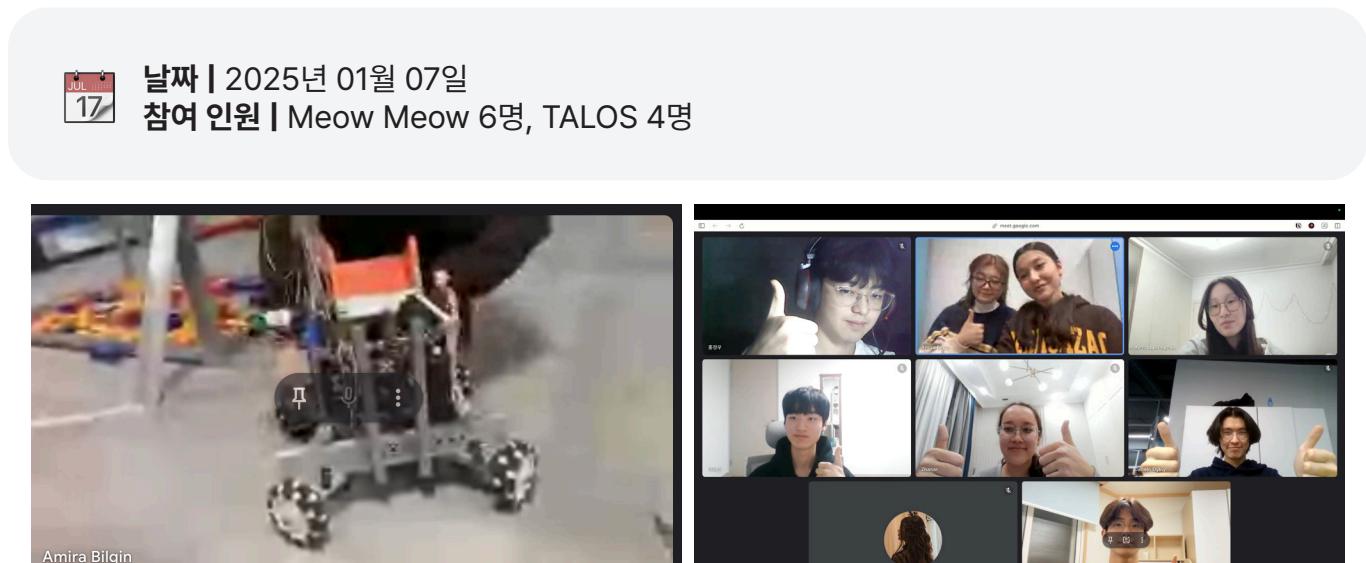
호주 FTC Nationals에서 Inspire Award를 수상한 동시에 Winning Alliance Captain이었던 14380 Blue BotBuilders와 1월 4일 미팅을 가짐. TALOS 팀에서 22 김수기, 22 최성빈, 23 김준성, 24 이소민, 24 이재빈이 참여함.

BlueBots 팀에서 미팅 전 로봇의 CAD 파일과 관련 영상을 보내줬으며, 이를 참고해 미리 질문을 준비함. Active intake의 설계를 구체적으로 설명해줌.

BlueBots에서 로봇의 각 부분을 보여주며 기능을 설명해주었으며, 각 부분에 대해 채택한 방식과 그 이유를 설명함. TALOS에서는 아직 로봇 제작을 시작하지 않은 관계로, 설계한 3D 모델을 보여주며 intake 방식을 구체적으로 설명함.

대회를 진행하며 어떠한 패널티를 조심해야 하는지 질문함. Ascent 시 다른 로봇에 닿지 않도록 조심해야 한다는 조언을 받음.

4) 24893 Meow Meow (카자흐스탄 FTC팀)



카자흐스탄의 Meow Meow 팀에서 각국의 FTC에 대해 교류하는 시간을 가지면 좋겠다고 연락을 하여, 1월 7일로 미팅 일자를 잡음. TALOS 팀에서 23 김준성, 23 홍정우, 24 이소민, 24 이재빈이 참여함.

왼쪽 사진과 같이 Meow Meow 팀에서는 X자 휠베이스를 사용한다는 설명을 들음. KRC나 FTC Championship에서는 본 적이 없는 휠베이스였기에 성능이나 조종 방식에 대해 질문함. TALOS의 탱크드라이브와 유사한 조종 방식을 설명하며, 각각의 장단점에 대해 의논함.

카자흐스탄에서는 총 8번의 Regional Competition이 있으며, 모든 경기에 자유롭게 참여할 수 있어 기회가 많다는 설명을 들음. TALOS에서는 우리 팀의 다양한 Outreach를 설명하고, 한국에서의 KRC 대회를 설명함. 또한, 2024년도에 참여한 세계대회 경험을 공유함.

5) 11129 Novi RoboTitans (미국 FTC팀)

날짜 | 2024년 10월 20일

참여 인원 | Novi RoboTitans 2명, TALOS 1명

6 MESSAGES

HJ Hyeyoung Jung

FTC 미국 미사건팀 Novi Robotitans 입니다.
To: xsorexia@gmail.com, Cc: Yeongjae Kwon, khosook@ksa.kaist.ac.kr

안녕하세요?

Novi Robotitans (11129)의 권영재 학생 엄마인 정혜영입니다.

김호숙 선생님께 TALOS팀의 outreach 담당자로 김준성 학생을 소개받아 연락드리게 되었습니다.

미시건 팀은 중학생들이 FTC 대회에 참여하도록 되어 있어서, 저희 아이는 8학년입니다.

현재는 build와 scouting을 담당하고 있어서, 아직 새로운 팀과의 미팅셋업이 완료되지 않아 제가 연락드리게 되었습니다.

저희 팀에서는 TALOS 팀과의 미팅에서 학생들만이 아니고, 코치들과 멘토들도 참여할 예정입니다.

시차가 있다보니, 첫 미팅의 일정과 앞으로 어떤 부분을 미팅을 하면 좋을지 우선 영재와 저, 준성학생이 사전 협의를 이루라도 매일로 논의하는건 쉽지 않을듯한데, 줌미팅을 하면 어떨까요?

미시건은 Eastern time zone 입니다. 현재 섬머타임 적용중이라 한국 오후 3시가 미시건 오전 2시입니다.

고통학생이라 많이 비�틀텐데, 언제 시간이 되실까요?

약 30분가량 일정을 집고, 팀간 첫 미팅 셋업을 위한 논의를 했으면 합니다.

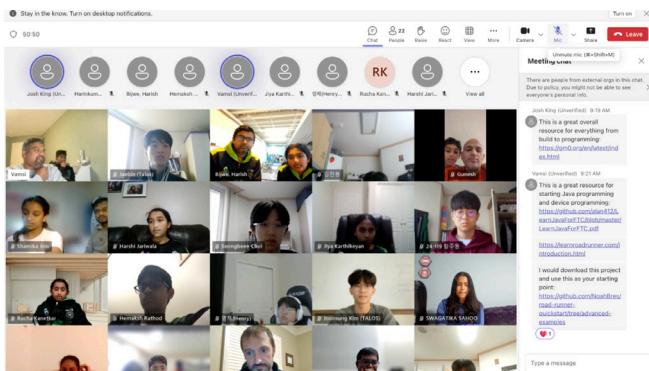


2024 FTC 세계대회에서 만났던 Novi RoboTitans 팀의 멘토분께서 한국의 FTC 활성화 및 TALOS 팀 멘토링을 목적으로 Zoom 미팅을 가질 것을 제안함. 앞으로의 미팅 계획 및 일정 등을 정리하고자 TALOS에서 23 김준성이 참여하여 10월 20일 첫 미팅을 가짐.

10월 26일 오전으로 두 번째 미팅을 계획하였으며, TALOS 팀에서 멘토링이 필요한 부분 등을 의논함. 프로그래밍 측면에서 FTC Dashboard 및 RoadRunner 등의 프로그램과 관련된 자료 및 빌딩 프로세스 등을 공유해 줄 것을 요청함. 한국과 미국의 대회 Judging 방식 차이를 공유함.

날짜 | 2024년 10월 26일

참여 인원 | Novi RoboTitans 15명, TALOS 7명



TALOS 팀에서 22 김수기, 22 최성빈, 23 김준성, 23 홍정우, 24 이재빈, 24 함주원, 24 이시우가 참여함. Novi RoboTitans 팀에서 코치 4명과 멘토 1명, 학생 10명이 참여함. 1시간 30분 정도동안 미팅이 진행됨.

각 팀 멤버들의 간단한 자기소개와 팀 소개를 진행함. Novi RoboTitans 팀에서 간단한 PPT를 준비하여 학생들이 돌아가며 발표함.

이전 미팅에서 요청했던 FTC Dashboard와 RoadRunner에 관련된 자료를 공유받음. 로봇 빌딩과 관련된 다양한 질문을 하고, 이에 대한 코치들의 답변을 받음. 또한, 다양한 방식 중 하나를 선택할 때 사용하는 PUGH Matrix 방식에 대한 설명을 들음.

6) 10001 RoboCavs Silver, 8479 RoboCavs Gold (미국 FTC팀)



날짜 | 2025년 01월 08일

참여 인원 | RoboCavs 5명, TALOS 4명



미국 매릴랜드의 RoboCavs 팀에서 TALOS의 웹사이트를 보고 한국의 FTC 프로그램과 아웃리치 활동에 대해 의논하는 세션을 가지자고 제안함. TALOS 팀에서 22 이규진, 22 유태우, 23 김준성, 23 홍정우가 참여함.

TALOS는 PPT를 통해 팀과 KRC를 소개했고, 이번 시즌에 만드는 로봇의 디자인을 간단히 설명함. RoboCavs 또한 캐드 파일을 보여주며 이번 시즌 로봇의 디자인을 소개해주었고, 간단하게 팀에 대해 소개한 후 멘토 제도 및 아웃리치 방법에 대한 자신들의 경험을 공유함.

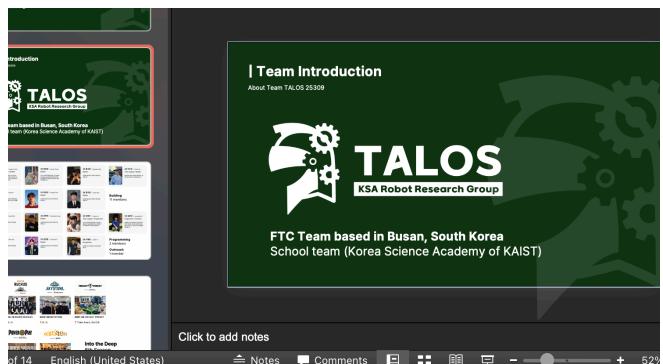
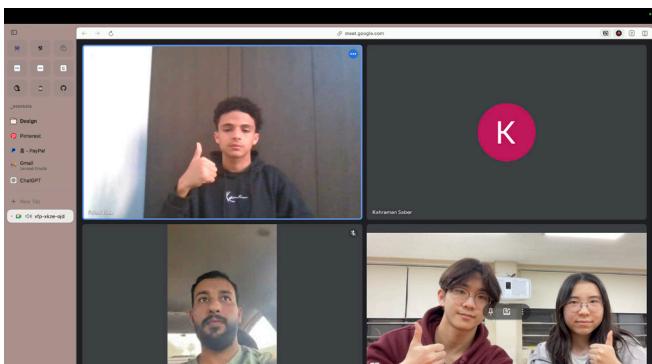
매릴랜드에서는 지역 대회에서 팀을 2개의 디비전으로 나누어서 진행한다고 함. 전반적으로 대회 진행 방식과 팀 운영 방식, 멘토 제도의 차이를 논했으며, RoboCavs 팀의 경우 팀원들의 부모님들이 멘토로 참여한다는 점이 인상적이었음.

7) 18422 Wizards Robotics (리비아 FTC팀)



날짜 | 2025년 01월 08일

참여 인원 | Wizard Robotics 4명, TALOS 2명



리비아의 Wizards Robotics 팀에서 TALOS의 인스타그램을 통해 이번 시즌의 로봇 디자인과 아웃리치 활동, 각국의 FTC 프로그램에 대해 의논하는 세션을 가지자고 제안함. TALOS 팀에서 22 이규진, 23 김준성이 참여함.

TALOS는 PPT를 통해 팀에 대한 소개와 이번 시즌에 만드는 로봇의 디자인을 간단히 설명함. FTC의 국내대회인 KRC에 대한 설명도 함. Wizards Robotics 팀은 자신들의 아웃리치 방법과 리비아에서의 지역 대회 방식을 설명함. 서로 로봇 디자인 및 팀 운영 방식 등 각 팀에 대해 궁금한 것을 질문함.

리비아에서는 지역별로 대회가 있어서 Wizards Robotics 팀을 포함하여 여러 번 출전하는 팀도 있다고 함. 또한 팀 내에서의 역할 분배는 프로그래머, 빌더, 아웃리치 담당의 비율이 각각 4명 정도로 같다고 함. Wizards Robotics도 Inspire Award를 받고 작년에 세계대회에 출전했었지만 우리와 디비전은 달랐다고 함.

8) 계획 및 결과

2023-24 CENTERSTAGE 시즌에는 다른 팀과 교류한 경험이 없었지만, 이번 시즌부터 활발히 교류하며 로봇 제작 뿐만 아니라 팀 운영 등 다양한 방면으로 도움을 받을 수 있다는 사실과 반대로 우리의 경험을 공유할 수 있는 좋은 기회라는 사실을 알 수 있었습니다. 24-25 시즌이 끝나기 전, 1월 17일 미국 휴스턴의 FTC 팀인 #12313 J.B. Alexander Robotics Team과 미팅을 할 예정입니다.



만난 사람들

39 명



교류한 시간

8 시간

총 7개의 해외 팀에서 39명의 사람들을 만나 교류하였습니다. 리비아의 Wizard Robotics과의 미팅에서 해당 팀은 매 시즌 50팀과 교류한다는 것을 듣고, 다음 시즌에는 보다 적극적으로 미팅을 진행해볼 예정입니다.

2. 국내 FTC 팀

1) 25317 O.R.B.I.T



날짜 | 2024년 08월 25일

참여 인원 | O.R.B.I.T 5명, TALOS 6명



TALOS에서 22 김수기, 22 강현빈, 22 장호원, 22 유태우, 23 김준성, 23 홍정우가 참여하였으며, O.R.B.I.T에서 5명 정도가 회의에 참여함.

O.R.B.I.T은 올해가 2번째 시즌이나, 작년 시즌에 거의 준비가 안 된 상태로 참여하여 올해 시즌을 어떻게 준비하면 좋을지 질문함. TALOS에서는 작년 시즌과 세계대회 참여 경험을 공유함.

2) 25360 R.O.G. United



2023-24 시즌 KRC에서 만났던 25360 R.O.G. United 팀에서 시즌 준비에 대해 질문함.

연구회장 23 홍정우가 연락을 받아 부품 구매 및 사용하는 센서 등에 대한 정보와 팀 코드를 공개하여 놓은 GitHub 등을 공유함.

3) 계획 및 결과

2024-25 시즌동안은 많은 한국 팀과 교류할 기회가 없었습니다. 그러나 작년 KRC와 Championship의 경험을 토대로 많은 정보를 공유할 수 있었고, 다음 시즌에는 더 많은 팀들에게 먼저 연락하여 미팅을 진행해볼 예정입니다.

TEAM TALOS
#25309

로봇공학연구회 KROS

ENGINEERING NOTE BUILDING

빌딩

해당 섹션은 로봇의 설계, 모델링, 제작 과정을 다룹니다.

목차 |

1. 로봇 설계 및 전략 개요
2. 설계 및 제작 과정

우이를 소개합니다.

시즌 캡틴 이름의 뒷글자를 따서 로봇 이름을 짓는 TALOS 팀의 전통에 따라, 캡틴 홍정우의 이름으로 로봇 “우이”를 명명했습니다.

우이는 독창적이면서도 빠른 이터와 팔 설계를 통해 효율적으로 점수를 낼 수 있는 구조를 가지고 있습니다.

Servo Power Module

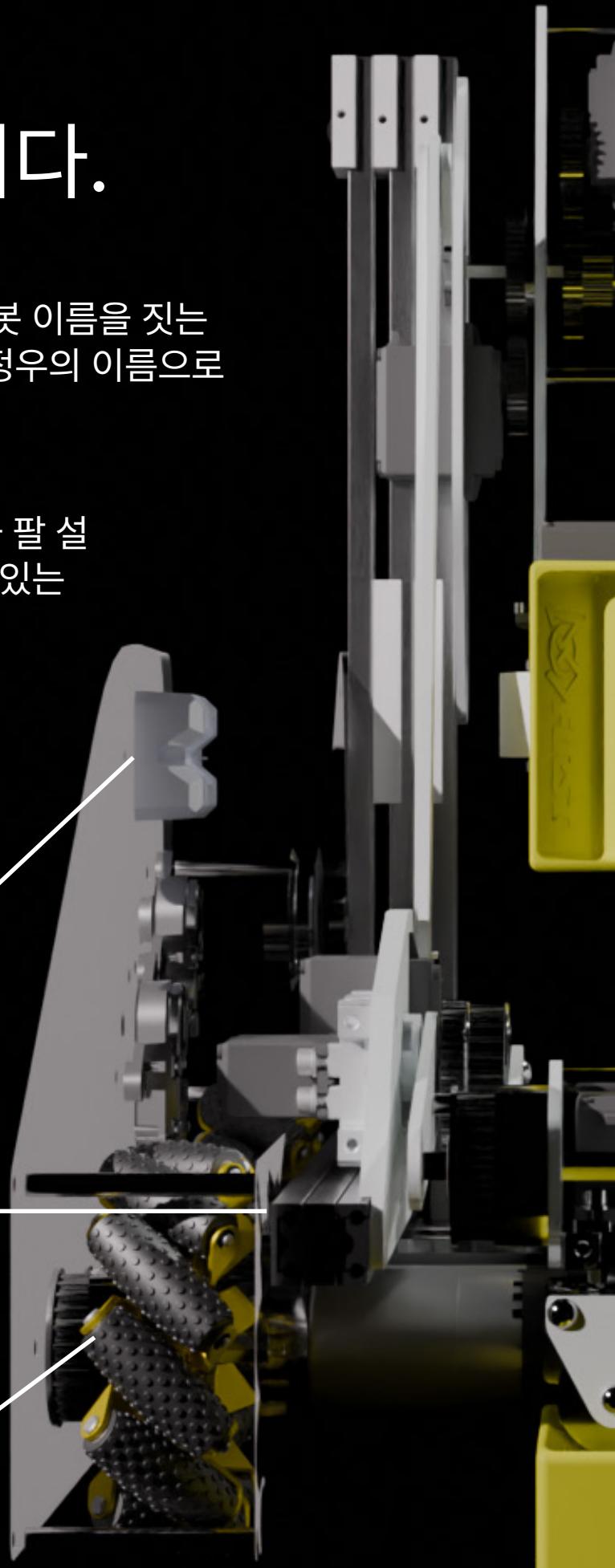
서보에 충분한 전력을 공급하는 동시에 과전류로부터 보호함.

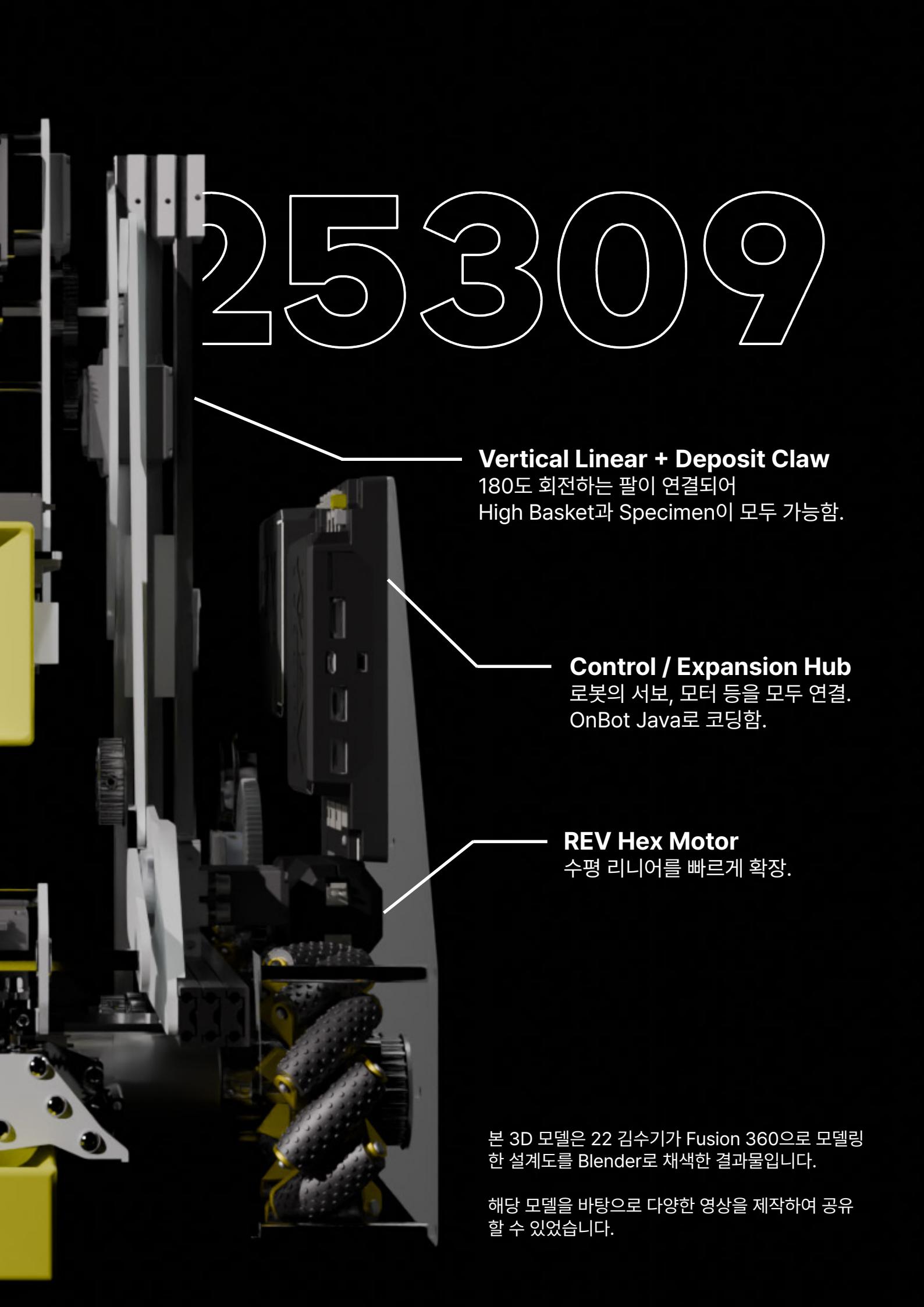
Horizontal Linear + Eater

집게 방식 active intake로 속도 극대화, Deposit Claw로 전달.

Mecanum Wheel

Belted Drivetrain 구조로 연결, 전후좌우 모든 방향으로 이동.





25309

Vertical Linear + Deposit Claw

180도 회전하는 팔이 연결되어
High Basket과 Specimen이 모두 가능함.

Control / Expansion Hub

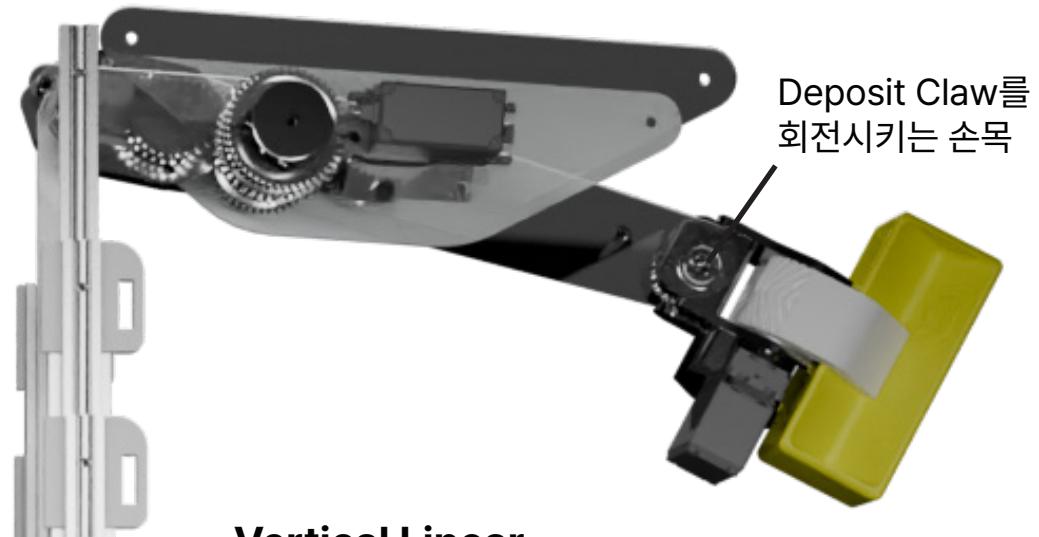
로봇의 서보, 모터 등을 모두 연결.
OnBot Java로 코딩함.

REV Hex Motor

수평 리니어를 빠르게 확장.

본 3D 모델은 22 김수기가 Fusion 360으로 모델링
한 설계도를 Blender로 채색한 결과물입니다.

해당 모델을 바탕으로 다양한 영상을 제작하여 공유
할 수 있었습니다.



Deposit Claw를 회전시키는 손목

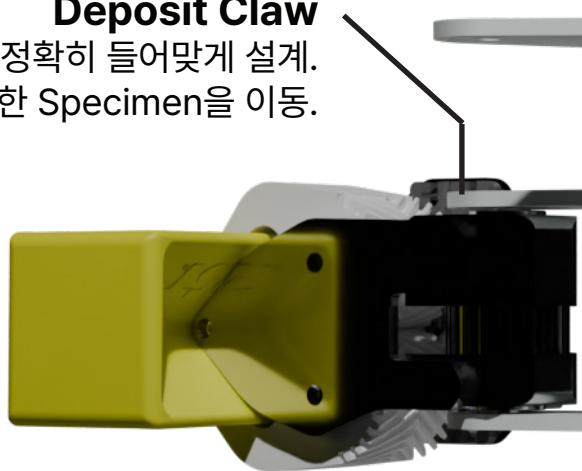
Vertical Linear

MISUMI 3단 수직 리니어.

실과 도르래를 이용, 축이 양방향으로 복원토크를 받게 하여 전개 후 자동으로 되돌아오는 구조.

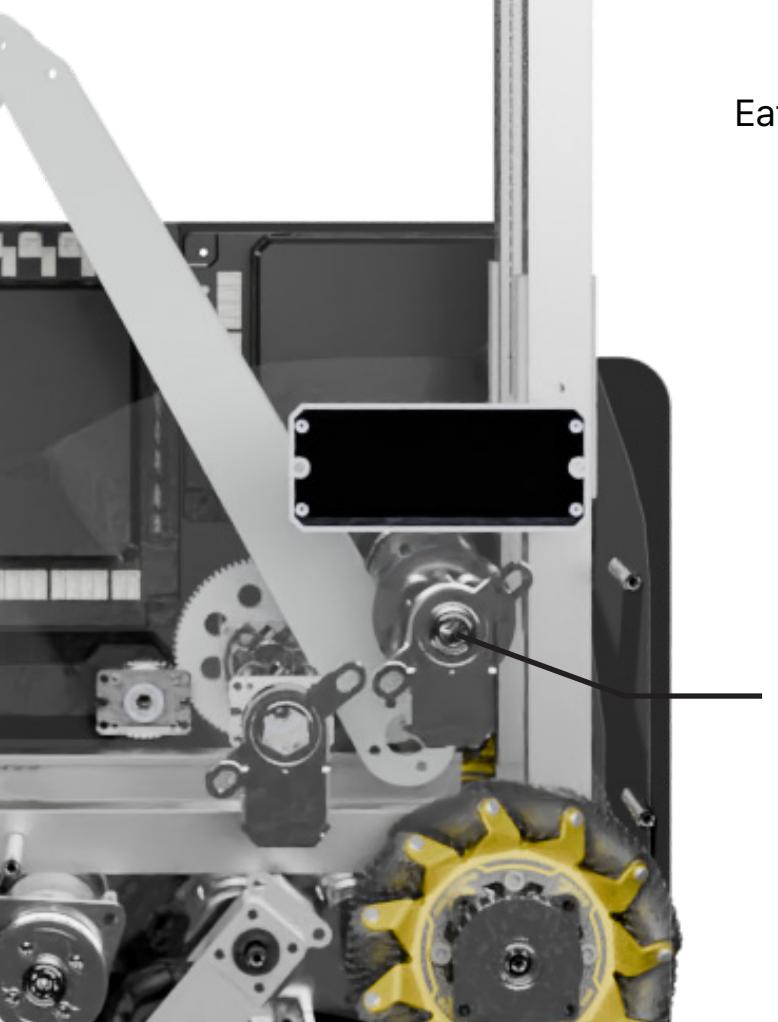
Deposit Claw

Specimen과 정확히 들어맞게 설계.
Eater를 통해 수집한 Specimen을 이동.



DC Motor (Gear)

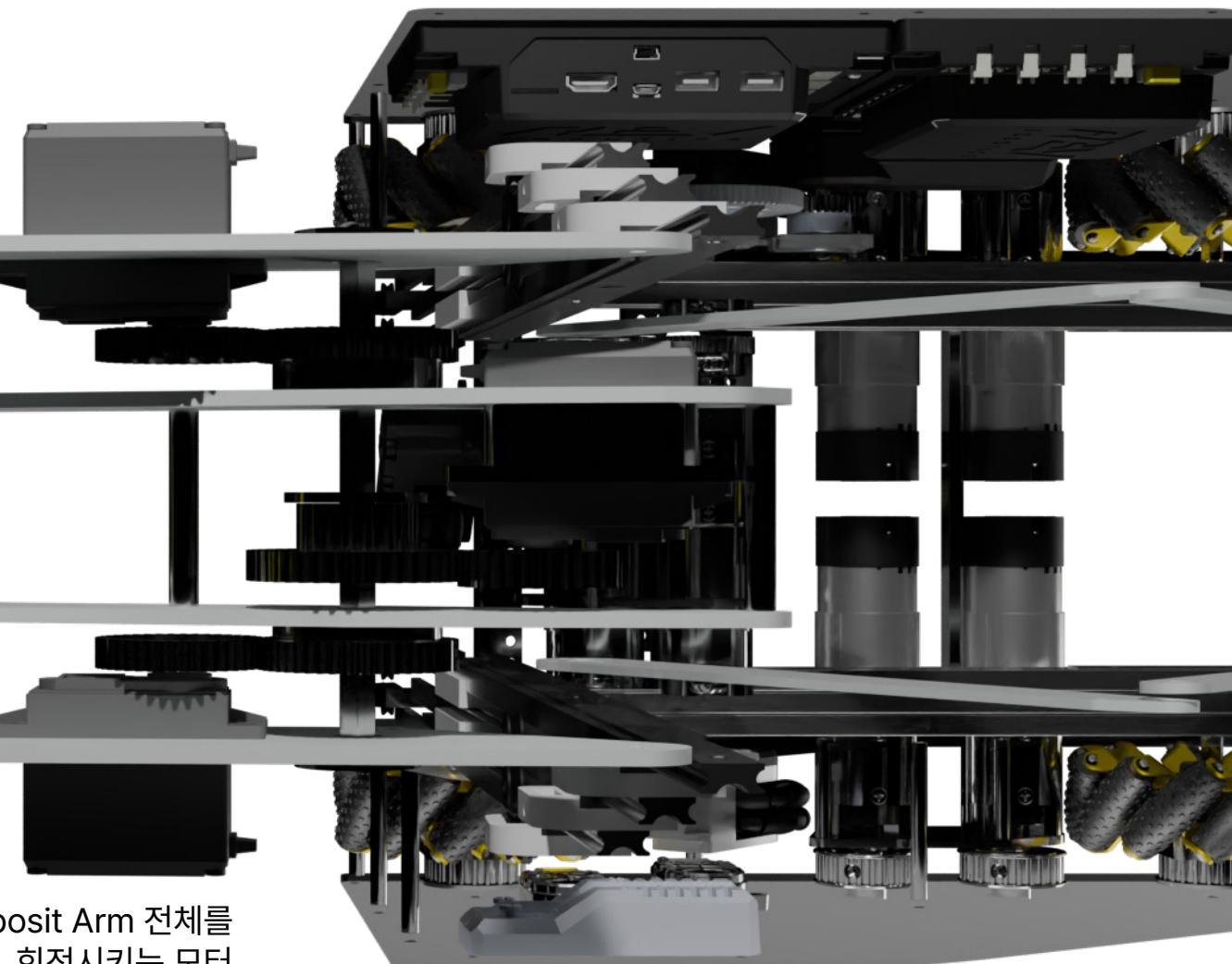
DC모터와 연결되어 도르래가 감긴 축을 빠르게 감아 리니어를 수직 방향으로 이동.



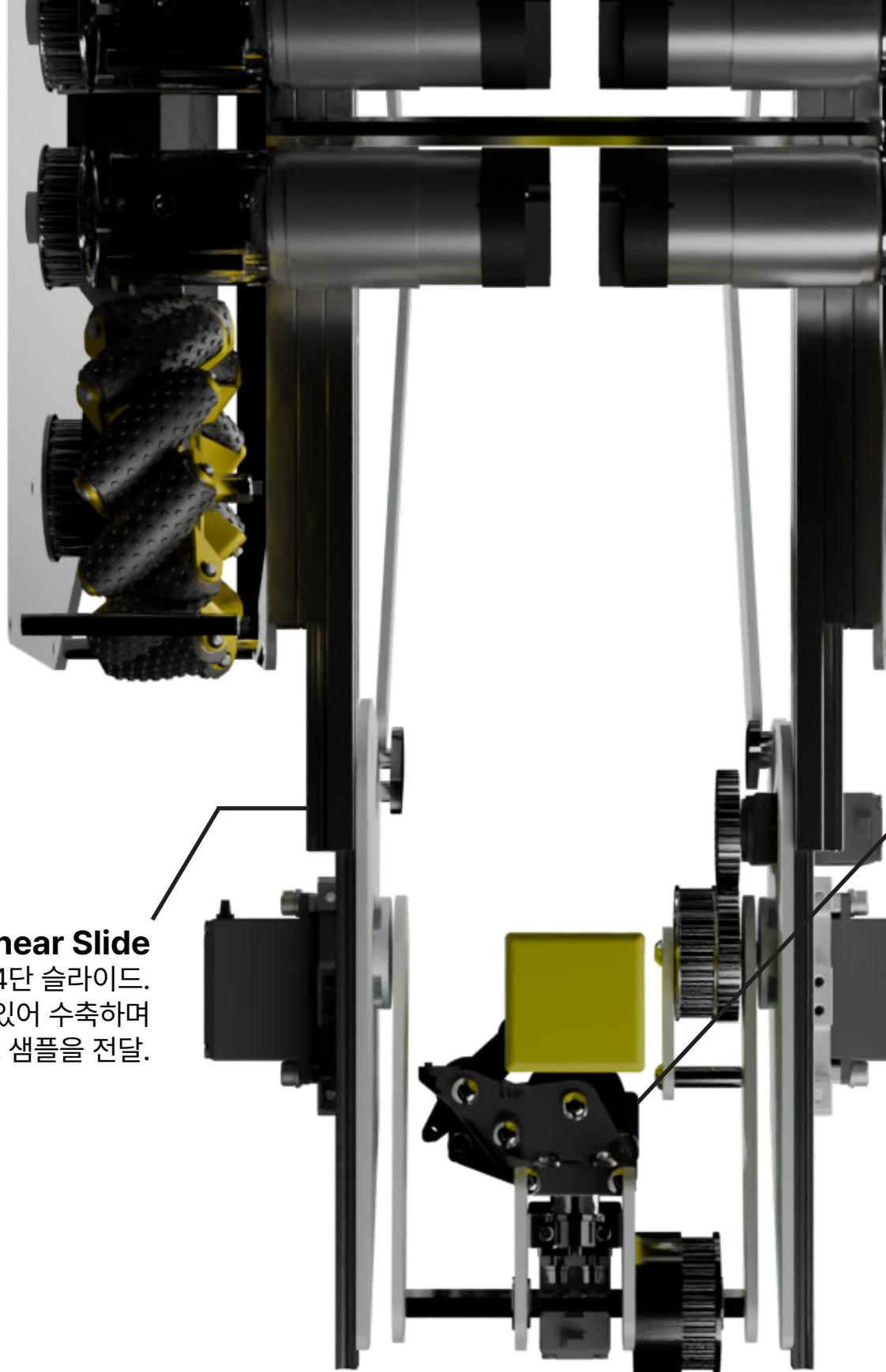
수직 리니어 + Deposit Arm

Deposit Arm

Deposit Claw가 연결되어 있음.
수직 리니어에 고정되어 서보 2개를
사용해 180도 회전.



Deposit Arm 전체를
회전시키는 모터



Horizontal Linear Slide

MISUMI 4단 슬라이드.
SugEater가 연결되어 있어 수축하며
Deposit Claw로 샘플을 전달.

수평 리니어 +
SugEater

Localization

Odometry를 사용해 경기장 위에서의 정확한 위치 확인 가능, Auto 주행 시 높은 정확도.

Belted Drivetrain

타이밍 벨트로 연결되어 구동되는 drivetrain.
모든 방향으로 움직일 수 있음.

SugEater

회전하는 롤러를 집게처럼 사용하여 Active Intake와 집게의 장점을 모두 살림.
3축 모두 자유롭게 회전할 수 있음.

Link Structure

REV Hex 모터에 연결되어
수평 리니어를 빠르게 확장시킴.

2024년 07월 02일 (화)

구분 | SUMMER_EN_BUILDING

책임자 | 23 홍정우



작성/편집 | 23 홍정우

작업 참여 | 22 유태우, 22 이규진, 23 김준성, 23 김진용, 23 홍정우 24 이소민, 24 이재빈

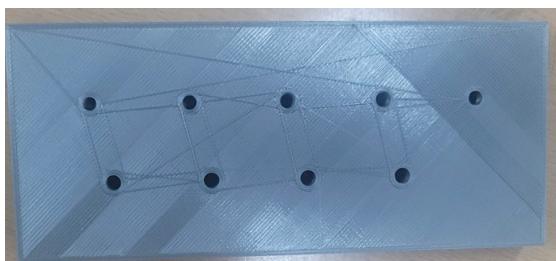
이전까지는 기존에 고빌다 혹은 다른 사이트에서 제공해준 철제 프레임을 기반으로 로봇을 제작하였었는데, 이번 대회에서는 로봇을 구체적으로 모델링하여 철제프레임을 쓰지 않고 로봇 차체를 만들어볼 계획임. 그렇기에 시간이 여유로운 여름방학 중에 학교에 잔류하여 여러가지 3d 프린팅과 아크릴 등을 활용하여 다양한 테스트를 해보았음.

1. 3D 프린팅 기어 테스트



3D 프린팅한 기어를 사용할 수 있을지를 확인해보기 위해 기어들을 모델링 한 후 프린팅 해보았음. 그러나 3D 프린터의 퀄리티가 좋지 않아 잘 되지 않았지만, 기어가 기본적으로 맞물리고 돌아가는 기본적인 역할은 할 수 있을것이라 판단하였음. 추가적으로 Double helix gear를 만들어보기로 계획함.

2. 3D 프린팅 M4 고정 테스트



3D 프린팅을 했을때 M4가 고정되게 하려면 어느정도의 구경을 가지는 구멍인지를 확인하기 위해 다음과 같은 모델을 프린팅했음. 하나씩 M4를 끼워본 결과 4.05~4.15mm 정도의 구경정도에 고정이되는 것을 확인할 수 있었음.

2024년 07월 03일 (수)

구분 | SUMMER_EN_BUILDING

책임자 | 23 홍정우



작성/편집 | 23 홍정우

작업 참여 | 22 김수기, 22 최성빈, 23 홍정우

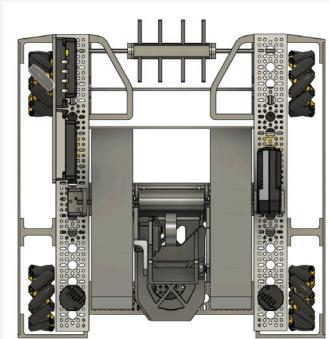
1. Prototype 차체 제작

새롭게 차체를 모델링해서 뽑아보기로 하였음. 기존에 사용하였던 고빌다 스타터 키트의 경우에는 DC모터와 기어를 사용하는데 이때 유격이 발생하는 등의 여러 문제점이 존재하였음.

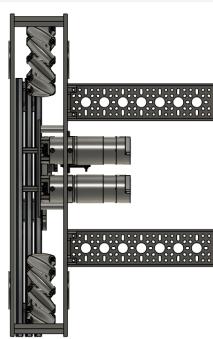
그렇기에 타이밍벨트 기반으로 DC 모터와의 연결 매커니즘을 바꿔서 설계해보았음. 이로 인해서 심한 유격을 줄일 수 있고, 공간 효율과 바퀴 축의 안정성을 향상 시킬 수 있었음.



CENTERSTAGE 차체와 비교!



△ CENTERSTAGE



△ Prototype

Prototype 차체는 Belted Drivetrain으로 제작하였습니다.

특징 / 장점 | 모터에 바퀴를 직접 연결하지 않고, 기어를 통해 연결하기에 차체 크기를 줄일 수 있으며 유격을 줄임. 타이밍 벨트로 연결하여 사용함.

2024년 07월 04일 (목)

구분 | SUMMER_EN_BUILDING

책임자 | 22 최성빈



작성/편집 | 22 최성빈

작업 참여 | 22 김수기, 22 최성빈, 23 홍정우

1. 아크릴 프린팅 베어링 고정 테스트



31.5mm 베어링이 들어가는 아크릴을 찾기 위해 다음과 같이 직경 31mm, 31.5mm, 31.7mm, 31.8mm, 32mm로 직경에 변화를 주어 확인해본 결과 31.7mm이 작은 유격으로 가장 안정적이었고, 이보다 작은 직경에는 들어가지 않았음.

2. 공업단지 방문



부산사상공업단지에 방문하여서 필요한 부품들을 구매하고 프레임을 외주를 맡길 수 있는지 확인해보았음.

2024년 07월 05일 (금)

구분 | SUMMER_EN_BUILDING

책임자 | 22 김수기



작성/편집 | 22 김수기

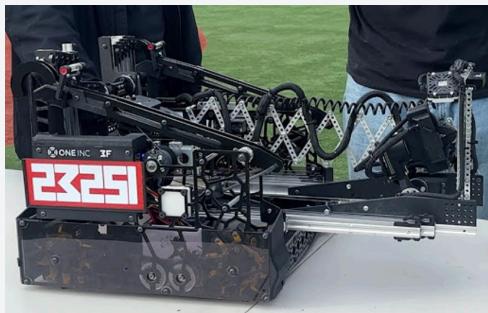
작업 참여 | 22 김수기, 22 최성빈, 23 홍정우

1. 시저 리프트 제작

링크 구조의 기본이 된다고 할 수 있는 시저리프트를 아크릴로 제작해보았음. 작동은 되었지만 많은 힘이 필요하였음.

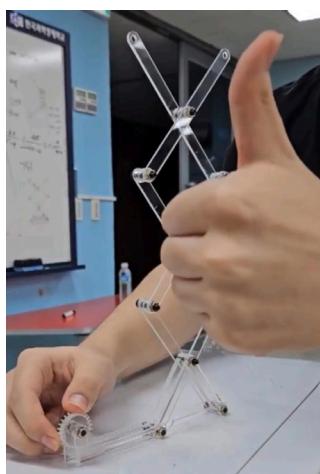


시저 리프트 (Caesar Lift)란?



X자로 연결된 구조물이며, 모터 등을 통해 수직 방향으로 확장됩니다. 많은 해외 팀들이 시저 리프트를 사용하여 수평, 수직 리니어를 확장하는 데 보조합니다.

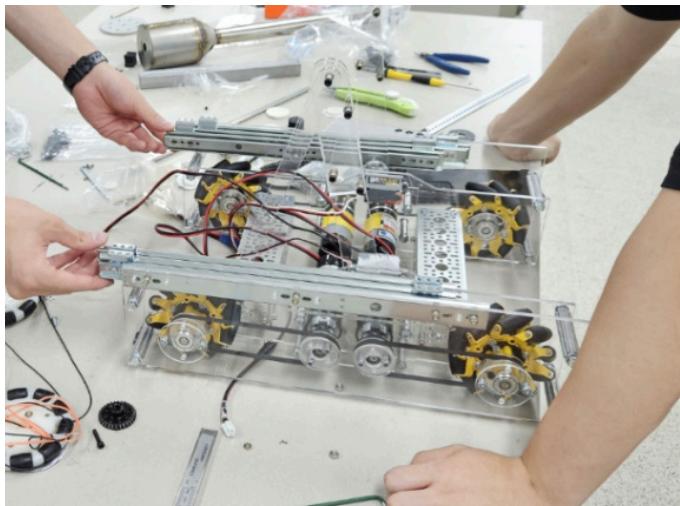
왼쪽은 23251 Triple Fault의 로봇입니다. 동력은 수평 리니어를 사용하며, 시저 리프트는 케이블 정리를 위해 보조만 한다는 사실을 알았습니다.



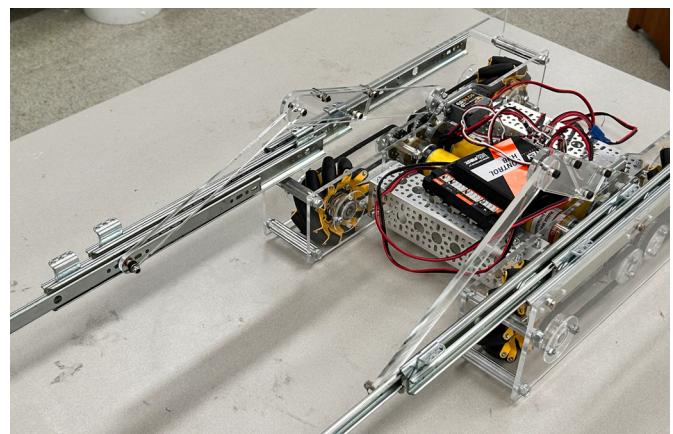
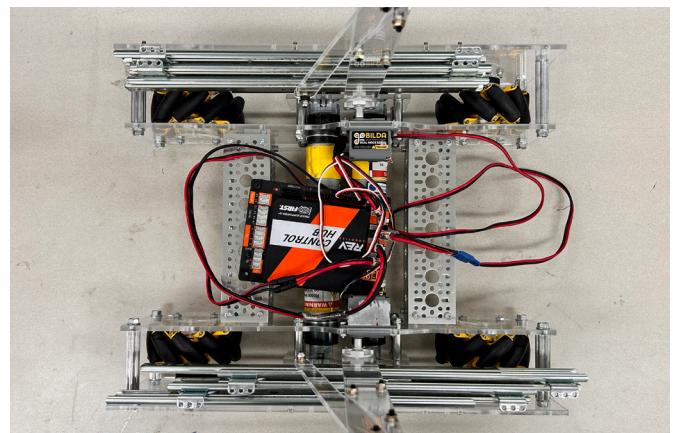
왼쪽과 오른쪽 사진은 각각 시저 리프트의 확장 전, 후의 사진. 케이블 정리를 할 때 어떤 식으로 고정하면 좋을지 실물을 보고 고민하기 위해 Rack and Pinion 구조를 사용해서 테스트해 보았다.

2. 프로토타입 차체 완성

타이밍 벨트를 기반으로 아래 차체부분을 제작 완료하고, 수평 리니어를 위에 부착하였음. 수평 리니어를 다음과 같이 서보와 아크릴을 통해서 만들게 됨.



타이밍 벨트를 기반으로 아래 차체부분을 제작 완료하고, 수평 리니어를 위에 부착하였음. 수평 리니어를 다음과 같이 서보와 아크릴을 통해서 만들게 됨. 완성된 사진은 아래와 같음.



2024년 12월 11일 (수)

구분 | SEMESTER_EN_BUILDING

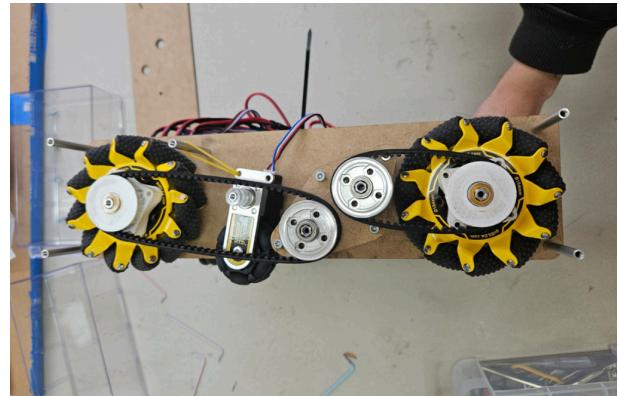
책임자 | 22 최성빈



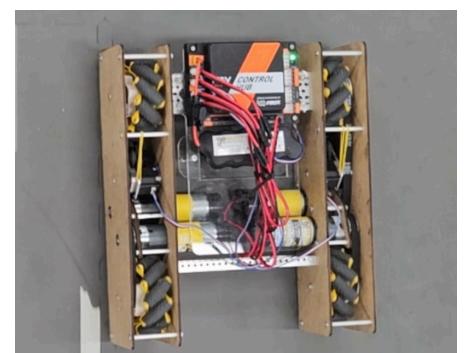
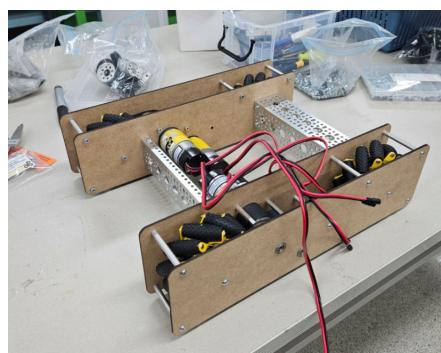
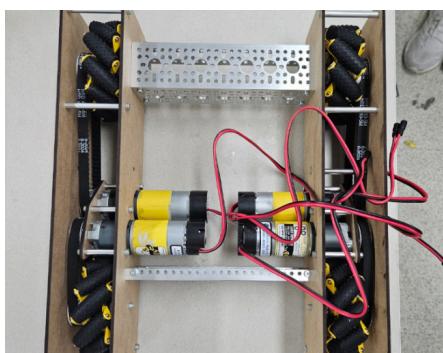
작성/편집 | 22 최성빈

작업 참여 | 22 김수기, 22 최성빈, 23 홍정우

1. RoadRunner 테스트용 차체 제작



CAD 파일을 기반으로 로드러너 테스트용 차체를 제작함. 로봇의 폭을 최대한 줄이기 위해 메카넘 휠, 타이밍 벨트 폴리, 베어링과 DC 모터 등 구동부를 만드는 데 필요한 부품을 밀도 있게 배치하여 옆판 사이의 거리를 56mm 정도로 줄일 수 있었음. 옆판은 mdf로 제작하였으며 옆판은 56mm Standoff로 고정함. 설계에 맞추어 구입한 닫힘형 타이밍 벨트를 사용하여 DC 모터의 동력이 바퀴로 전달되도록 했으며 오도메트리는 일차적으로 차체를 조립해본 뒤 나중에 추가함. 오도메트리 세트에 사용되는 용수철 대신 고무줄을 사용하여 오도메트리가 바닥과 안정적으로 밀착될 수 있도록 함. 그리고 옆판끼리는 U 채널 두 개를 사용하여 연결하였으며 연결한 U 채널 위에 아크릴로 Control hub와 배터리가 들어갈 공간을 만들었음. 차체 완성 후에는 제대로 동작하는지 확인해보았고 잘 움직임.



2025년 01월 06일 (월)

구분 | WINTER_EN_BUILDING

책임자 | 23 홍정우



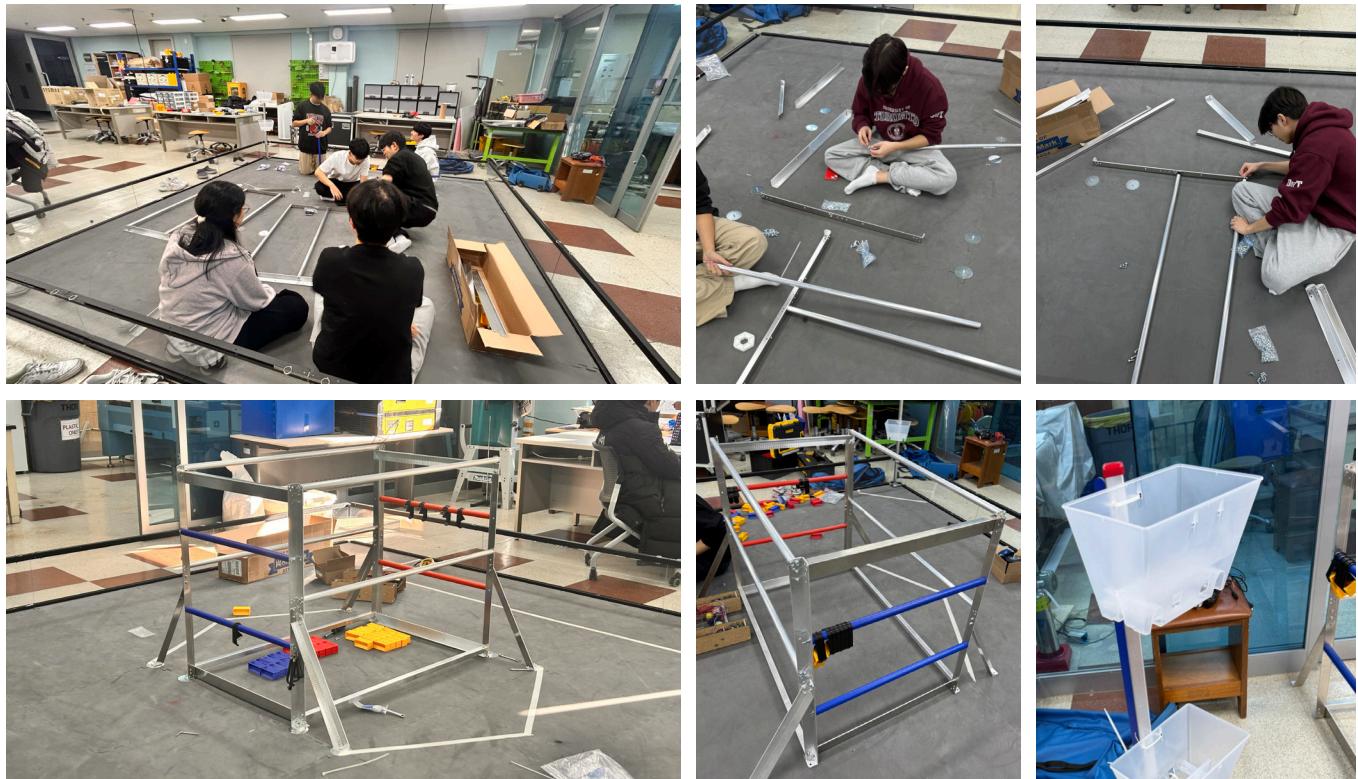
작성/편집 | 23 홍정우

작업 참여 | 22 유태우, 22 이규진, 23 김준성, 23 김진용, 23 홍정우 24 이소민, 24 이재빈

1. 경기장 제작

1월 6일 잔류를 시작하였으며, 먼저 도착한 일부 인원끼리 Into the DEEP 경기장의 조립을 시작함. Andy-Mark에서 Partial game set를 사전에 주문하여, 배송된 부품들을 조립함.

6*6 크기로 바닥 타일을 깔고 주변에 벽을 세움. 나머지 바구니와 가운데 잠수정을 조립하여 설치함.



2024년 01월 07일 (화)

구분 | WINTER_EN_BUILDING

책임자 | 23 김준성

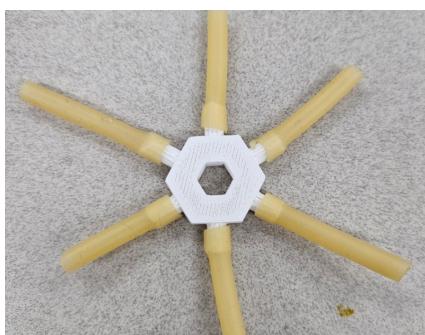
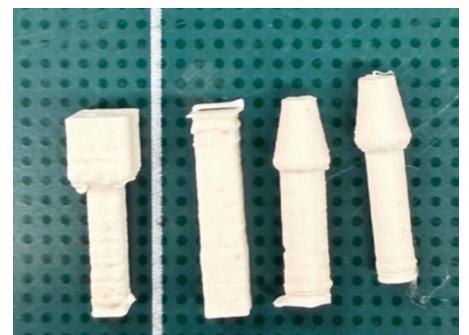
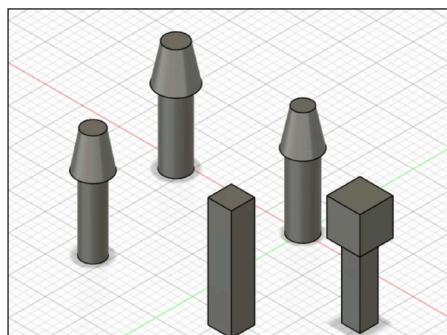


작성/편집 | 22 최성빈, 23 홍정우, 24 이소민

작업 참여 | 전체

1. 롤링 Eater 제작

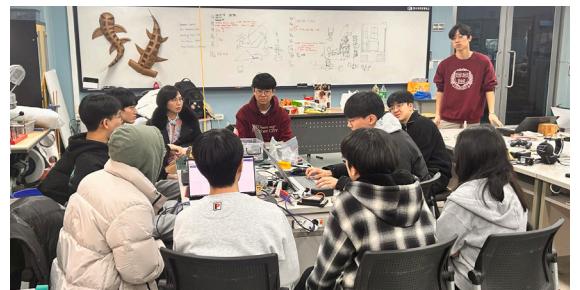
기존의 해외팀들 중에서 로터와 비슷한 형식으로 specimen을 가져가는 모델이 있었고, 로터를 막대에 surgical tube를 붙여서 만들기로 계획하였음. 막대에 surgical tube가 고정되려면 돌기가 필요하다고 판단하였고, 다양한 모양의 돌기를 제작해보았음. 기본적으로 4mm 지름을 가지는 막대에 돌기형태를 다르게 하여서 적합한 막대기 모양을 찾아보았음.



가장 오른쪽의 사진에서 순서대로 1. 정육면체 모양의 돌기 2. 돌기가 없는 형태 3.아래는 5mm, 위는 2mm 의 돌기 4. 아래는 4.5mm 위는 2mm 의 돌기임. 왼쪽의 사진과 같이 Surgical tube를 각 막대기에 끼워본 결과 돌기가 없이 잘 들어간다는 것을 알아내었음.

2. 전체 회의

연구회원 모두가 학교에 도착하여 로봇 제작 일정이 대해 김호숙 선생님과 회의를 진행함. 업무를 분담하고 매일 오전 10시에 작업을 시작하기로 결정함.



2025년 01월 08일 (수)

구분 | WINTER_EN_BUILDING

책임자 | 22 김수기



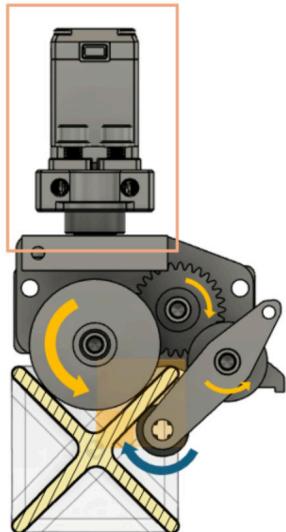
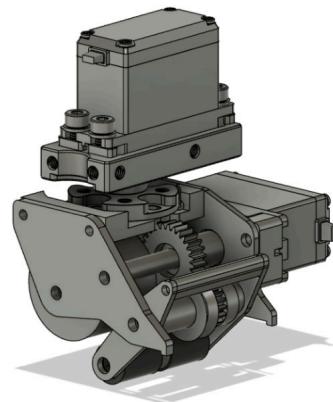
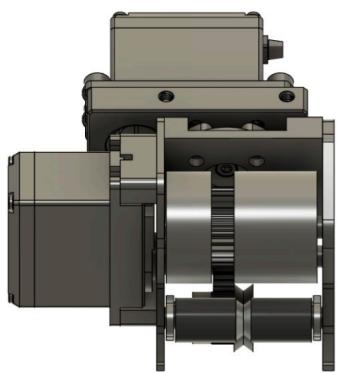
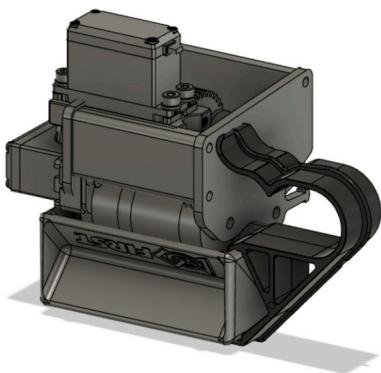
작성/편집 | 22 이규진

작업 참여 | 22 김수기, 22 이규진, 22 강현빈, 22 유태우

1. SugEater (수기 Eater) 제작

1) Eater 설계

이터가 여러 각도와 관련없이 블록을 잡을 수 있도록 롤러를 통해 블록을 빨아들이는 원리로 이터를 설계함. TALOS 팀원인 김수기가 이터를 설계하여 수기 이터라고 명명함.



서보모터와 3개의 기어를 사용해 하나의 동력으로 블록을 잡는 두 방향의 힘을 만들도록 설계함. 3D프린터로 기어를 제작할 경우, 유격이 발생하여 기어가 서보의 스피드에 따라 헛도는 경우가 발생하기 때문에, 이를 보안하기 위해 장력을 사용함.

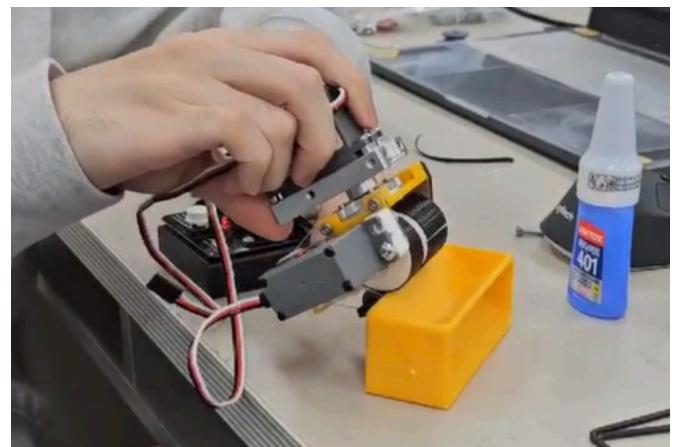
이터의 앞부분이 벌어지면 부착된 고무줄이 장력을 통해 다시 이를 잡으면서 블럭을 잡는 안정성을 추가함. 또한, 고무줄의 장력을 통해 일정한 유격을 유지함. 서보 모터에 연결된 축을 단단하게 만들기 위해 반대 부분에 standoff와 베어링을 연결함.

2) 이터 제작

이터를 제작하는 과정에서 서보와 연결되는 3D프린팅 모델 부분이 서보와 맞물리는 부분이 잘 맞지 않아서 다시 모델링을 진행함. 기존 서보 톱니 부분의 1.0배, 1.05배, 1.1배, 1.2배로 맞물리는 부분을 제작하였고, 서보에 끼워본 결과, 1.05배가 가장 적합하다고 판단하고 교체하였음.

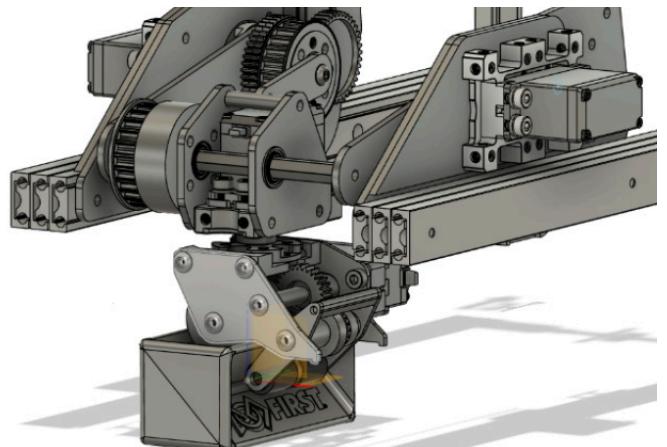
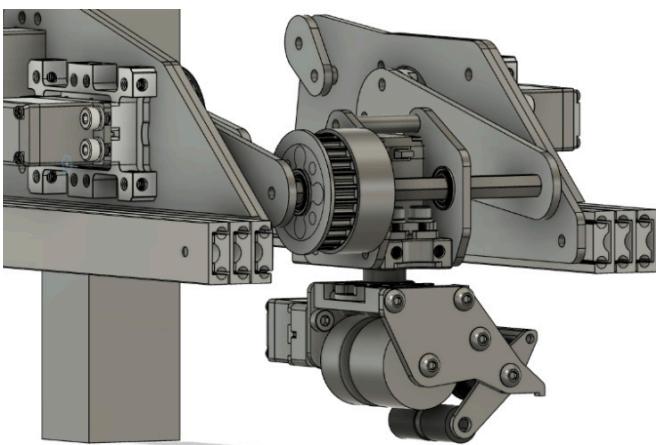


해당 이터를 작동시켜본 결과, 살짝 기울어져 있는 샘플이라도 쭉 밀면 샘플을 안정적으로 잡을 수 있음을 확인함. 다만, 위에서 내려찍으며 샘플을 먹을 수 있을 것으로 기대하였지만, 실제로 위에서 샘플을 먹기 위해선 비교적 위치의 제약을 받는다는 것을 확인함. 따라서 teleOP 때 빠른 동작을 위해서는 미는 방식으로 픽셀을 먹어야한다는 결론을 도출함. 따라서 잠수정 내부의 샘플을 먹기 위해서 이터의 높이를 조절할 수 있는 팔이 필요함을 인지했고, 이를 설계하기 시작함.



3) 이터 팔 설계

이터와 수평 리니어를 연결하는 설계도를 작성.



서보에서 두개의 freedom을 가지도록 설계를 했기 때문에 수평 리니어를 달아주면 먼 거리를 이동하지 않더라도 샘플을 집을 수 있다는 장점을 가지고 있음. 수평 리니어와 이터를 연결하는 과정에서 총 3개의 서보 모터를 사용하는데, 대칭적으로 있는 서보를 통해 이터의 손목을 움직이는 동력으로 사용하고, 하나는 팔을 움직이도록 설계함.



작성/편집 | 22 최성빈
작업 참여 | 22 최성빈

2. 에폭시 레진을 이용한 mdf 경화 테스트

MDF를 에폭시 레진에 적셔서 안쪽까지 스며들게 만든 다음 경화시키면 MDF의 단점인 낮은 내구성 문제를 해결할 수 있는지 확인하기 위해 진행함 (테스트 목적).

원래는 알루미늄 차체 키트를 사용하여 로봇의 구조를 만들었으나 이번에는 자체 제작 차체를 사용하게 되었으므로 레이저 커터를 사용하여 옆판을 만들게 됨. 그런데 아크릴은 충격에 쉽게 깨지고, 단단한 집성목이나 원목은 5T 미만으로 얇게 가공이 어려우며, 알루미늄 판 가공은 CNC가 필요한데 장비가 없으므로 결국 mdf를 사용하게 됨. 차체 옆판으로 사용하는 것에 있어서 그렇게 큰 문제는 없지만 지속적인 부하로 인해 휘거나 충격에 의해 일부가 뜯겨나갈 수 있으므로 두께를 바꾸지 않으면서 경도만 올릴 방법을 궁리하게 됨.



일단 10cm*10cm 3T mdf를 5장 준비하여 그중 3장에 주제와 경화제를 3:1 비율로 혼합한 에폭시 레진을 박스 조각을 이용하여 고르게 펴바름. 작업 전후 무게 차를 측정하여 10cm*10cm 한 면을 코팅하는데 사용된 레진의 양을 계산해보니 대략 1g 미만이 도출됨. 걱정되는 것은 단순히 겉에 레진을 바른다고 안쪽까지 레진이 스며들지는 잘 모르겠다는 것. 레진이 경화되는데 보통 12~24시간이 소요되므로 다음 날에 결과를 확인함.

경화



경화



결과적으로 경화한 MDF와 경화하지 않은 MDF 모두 같은 강도의 충격에 부서진다는 것을 확인할 수 있었음. 즉, 추후 MDF를 사용할 때 경화하지 않고 사용하기로 결정함.

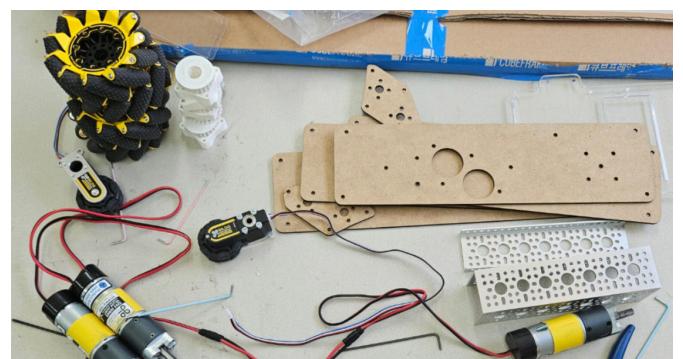


작성/편집 | 23 김준성

작업 참여 | 22 이규진, 22 최성빈, 23 김진용, 23 김준성, 23 홍정우

3. 테스트용 차체 분해

여름 잔류기간 동안 제작했던 차체를 분해하여 정리함. 부품도 종류별로 정리하였으며, 디자인 한 모델을 기반으로 새로 제작할 수 있도록 준비함.

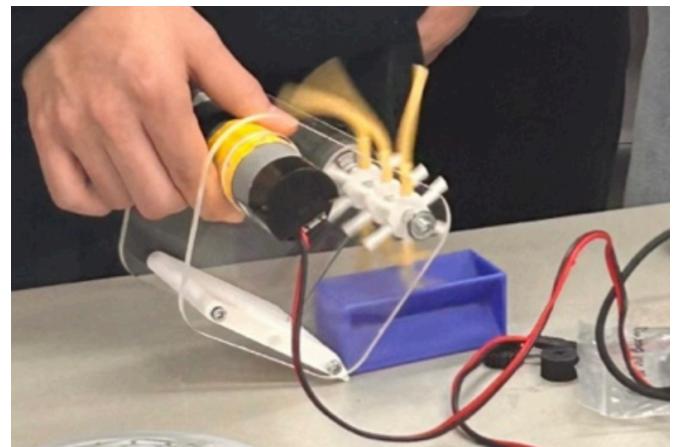
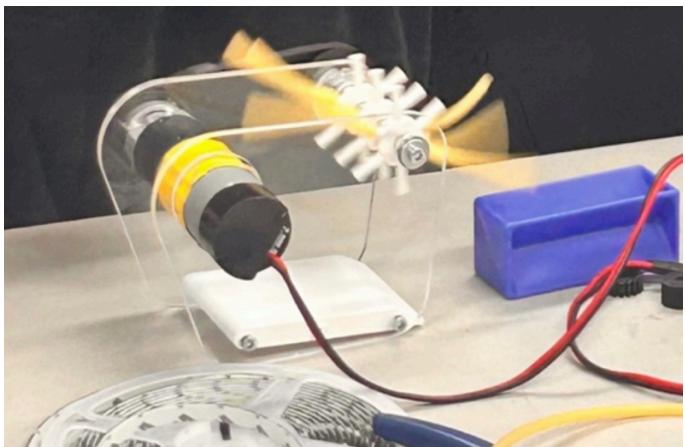




작성/편집 | 24 이소민
작업 참여 | 24 이소민, 24 함주원

4. 롤링 Eater 제작

Surgical tube에 적합한 막대기를 찾고 그후 롤링이터를 제작함. 롤링이터는 DC모터와 타이밍 벨트를 이용하여 만들었음.



롤링 이터는 양옆의 판을 5T 아크릴을 사용하여 제작했고 이외의 부품들은 3D 프린터로 출력 함. Specimen이 롤링이터에 의해 끌려가는 것은 확인했지만 바닥에 부딪혀 specimen이 올라가지 못함. 그렇기에 바닥 부분에 specimen이 경사로를 올라갈 수 있는 추가적인 장치가 필요하다고 판단됨.



SugEater과 비교!

장점 | SugEater는 집게를 사용한 방식이기에 정확한 위치에 집어야 합니다. 반면 Rolling Eater와 같은 active intake 방식은 sample 근처에만 있으면 비교적 부정확한 위치에서도 집을 수 있다 는 장점이 있습니다. 또한 서보 모터 대신 DC 모터를 사용하여 속도를 훨씬 높일 수 있습니다.

단점 | 그러나 집을 수 있는 방향이 세로로 한정되어 있으며, 현재로는 바닥에 부딪힌다는 문제가 있습니다.

2025년 01월 09일 (목)

구분 | WINTER_EN_BUILDING

책임자 | 23 홍정우

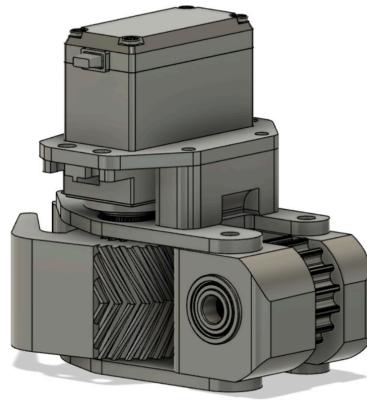
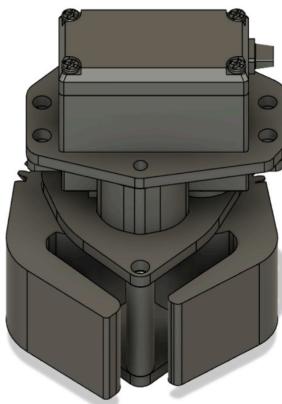


작성/편집 | 22 최성빈, 22 장호원

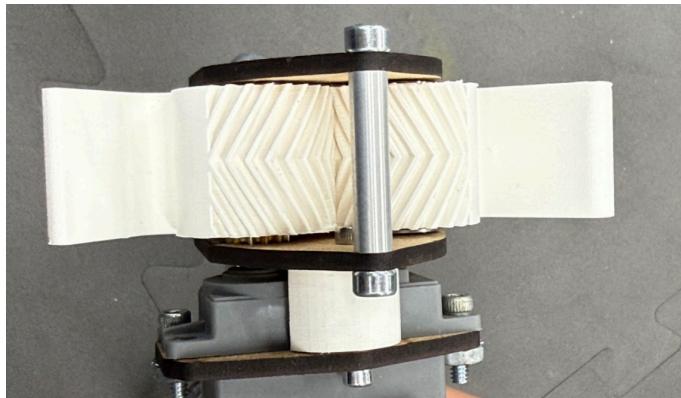
작업 참여 | 22 이규진, 22 최성빈, 22 장호원, 22 강현빈

1. Deposit 집게 제작

롤러를 통해 잡은 샘플을 다시 잡을 deposit 집게를 제작함. 톱니 구조를 활용해 하나의 서보 모터로 집게를 구현함. 샘플이 들어가 있는 구조를 활용해 정확히 잡을 수 있는 구조를 설계함. 이후 팔에 연결할 기어 또한 철사를 통해 고정함.



기어는 helical 기어를 사용해 반대로 된 헬릭스 각도에 의해 축방향 힘이 상쇄되어 더 안정적이고 빠르게 서보모터 작동이 가능함.



위 사진과 같이 조립을 완료함.



작성/편집 | 23 김준성

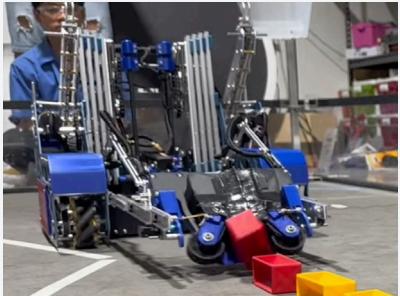
작업 참여 | 23 김준성, 23 김진용, 23 류승완, 23 홍정우

2. 사이드 Eater 제작

롤러를 사용하는 해외 팀들의 영상을 참조하여 이터를 제작함. 다른 이터에 비해 훨씬 빠른 속도로 intake할 수 있을 것이라고 기대하였으며, 운전자가 보이지 않는 곳에서 조종하기에 가장 좋은 방법이라고 생각하였음.

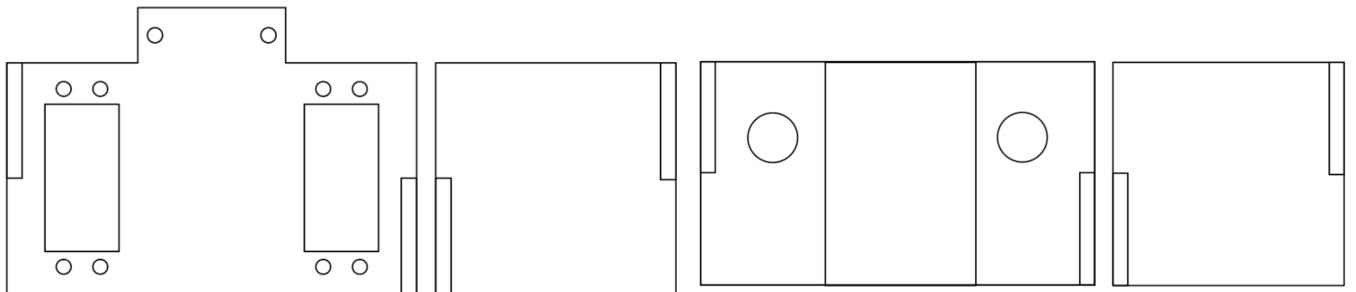


참고한 해외 팀 영상

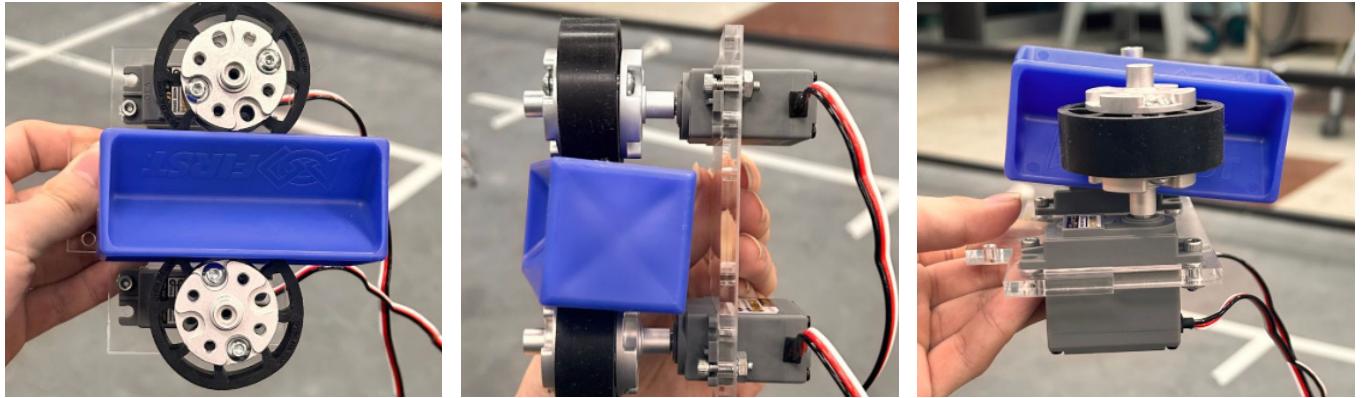


왼쪽부터 순서대로 14380 Blue BotBuilders, 14872 Orbit Knights, 18228 Red BotBuilders의 active intake. 이중 Orbit Knights의 이터를 참고하였음.

첫 시도에서 서보는 GoBilda의 Torque 서보를 Continuous Rotation으로 설정하여 사용하였으나, 속도가 느려 Superspeed 서보로 변경하였음. Adobe Illustrator로 아크릴 도면을 그려 출력하였으며, 서보를 끼워 테스트해보았음.



아크릴 도면은 위와 같음.



위의 사진과 같이 샘플을 집은 후 모터의 회전을 멈춤으로써 집은 상태를 유지할 수 있었음. 컬러 센서는 코드가 완성되지 않은 관계로 테스트해보지 못하였음.

샘플을 먹는 과정은 아래와 같음. 제작 후 몇 번에 걸쳐 테스트해본 결과 가로와 세로 등 모든 방향에서 샘플을 집을 수 있음을 확인함. 샘플 여러개가 서로 이웃하여 모여있는 경우에도 잘 작동함을 확인함.

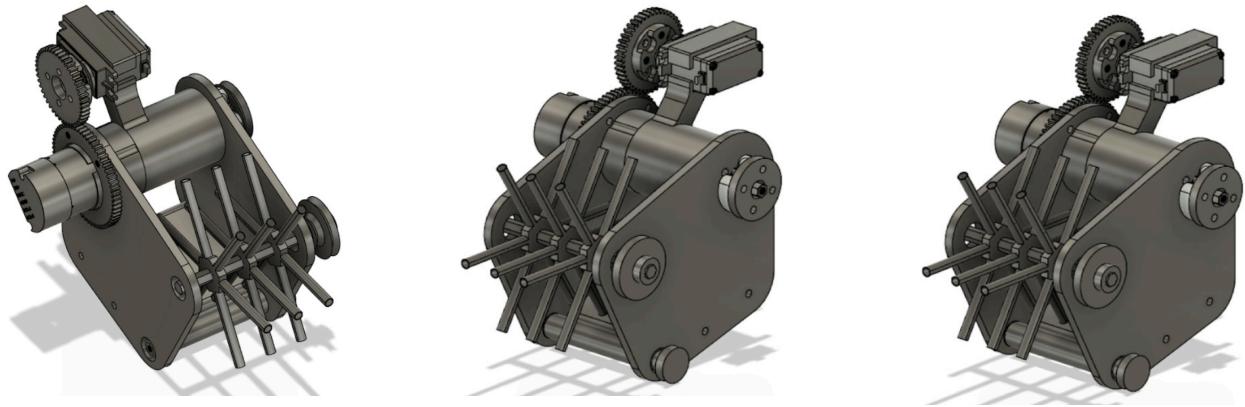


작성/편집 | 24 이소민
작업 참여 | 24 이소민

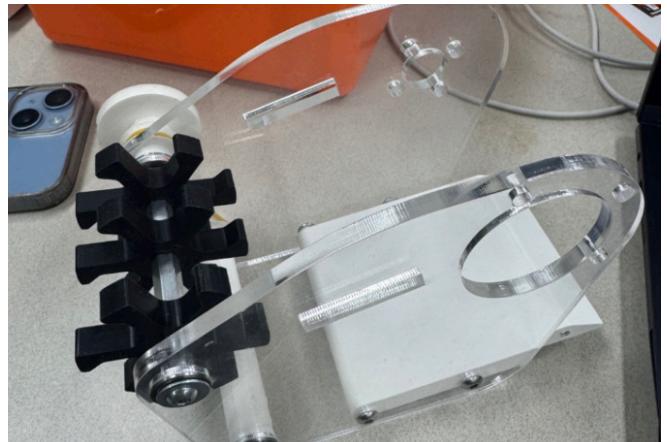
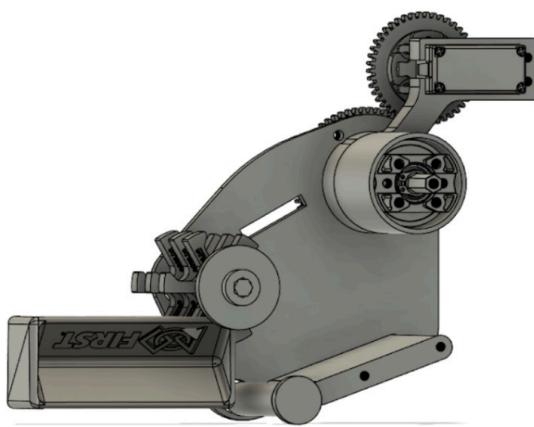
3. 롤링 Eater 개선

앞서 만들었던 롤링 intaker가 specimen을 먹지 못해서 앞에 롤러를 붙여 specimen을 먹는 방법을 고안하여 제작을 시도함. 또한 기존에 DC 모터에서 동력을 전달하기 위해 사용한 타이밍 벨트 방식이 과도하게 공간을 차지해, 고무줄을 이용한 풀리형태로 설계를 수정함.

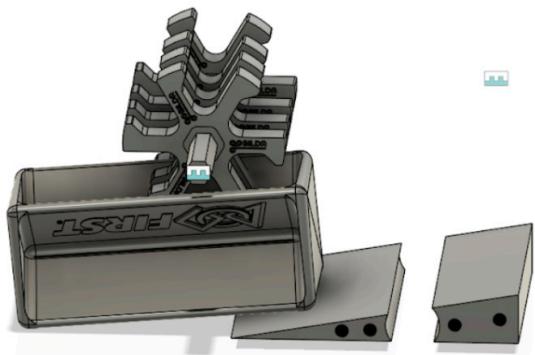
Intaker의 회전을 위해서 DC 모터를 축으로 회전하는 서보모터를 이용한 회전부를 설계함.



제작하여 실험해보니, 고무 튜브를 사용한 intaker는 롤러가 있어도, 롤러의 높이를 넘어 specimen을 잡기 위한 힘을 제공하지 못함. 때문에 GoBilda 사의 intake 전용 OD Wheel을 사용하여 재설계함.



하지만 본 모델도 intaker wheel이 롤러의 높이를 높기 위한 충분한 힘을 제공해주지 못함. 이를 해결하기 위해 다른 사례들을 찾아본 후, 전면부에 롤러를 없애고 매우 낮은 경사형태로 바꾸어 specimen이 쉽게 intake 되도록 수정함. Intaker에 들어간 specimen이 로봇으로 운반되기 위해서 경사 중간에 롤러를 배치함.





작성/편집 | 23 김진용

작업 참여 | 23 김진용, 23 류승완, 24 함주원

4. 3단 리니어 가이드 조립

앞서 제작한 롤러 이터가 장착된 헤드가 앞뒤로 자유롭게 움직이며 샘플을 다음 단계로 전달할 수 있도록 하기 위해 MISUMI사의 수평 리니어를 3단으로 조립하여 제작함. 리니어의 전개는 아래 그림과 같이 팔 막대를 달고 이를 모터로 회전시키는 방식. 이 때 리니어를 펼치고 접을 막대를 리니어와 연결하기 위해 연결부를 따로 아크릴 판을 이용하여 제작함.



2025년 01월 10일 (금)

구분 | WINTER_EN_BUILDING

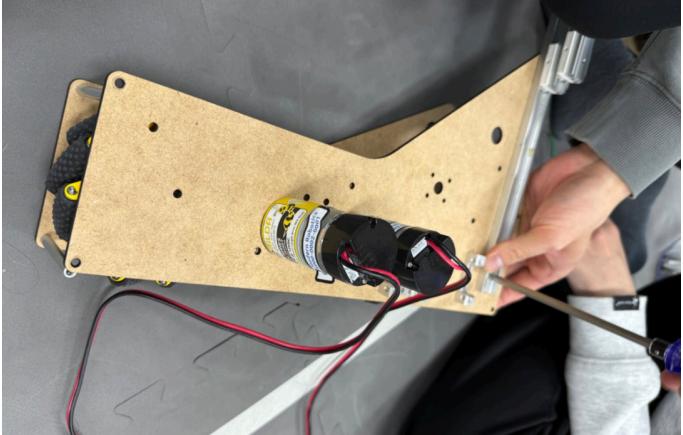
책임자 | 22 김수기



작성/편집 | 23 홍정우
작업 참여 | 전체

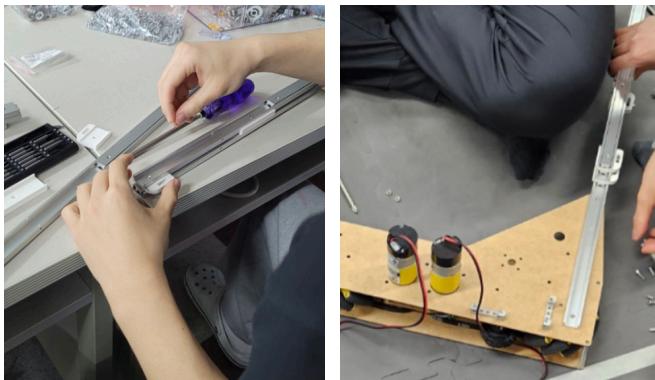
1. 차체 재구성

RoadRunner 테스트를 위해 만든 차체를 분해하고, 단순히 주행의 역할만 하는 차체에서 더 발전시켜 수평, 수직 리니어를 부착할 수 있는 공간을 추가하여 새로 차체를 출력 및 조립함. 실제로 보다 좁은 폭을 위해 차체에서 남는 공간이 최대한 효율적으로 사용함. 구동하는 DC 모터를 세로로 두어 바퀴에 바로 연결했을 때, 수평 리니어가 들어갈 공간이 없었음. 이를 인지하고, 모터를 가운데에 두어 타이밍 벨트로 동력을 전달하는 차체를 구축함.



완성된 결과를 보면 타이밍 벨트를 통해 모터와 바퀴가 직접 연결되지 않아도 굉장히 폭이 좁게 설계되었기 때문에, 접시 모양 나사를 사용해 리니어를 차체에 밀착시켜 제작함. 정확한 Auto를 위한 오도메트리도 standoff에 고무줄을 걸어 설치함. 이를 통해 굉장히 작은 사이즈의 차체를 제작할 수 있었음.

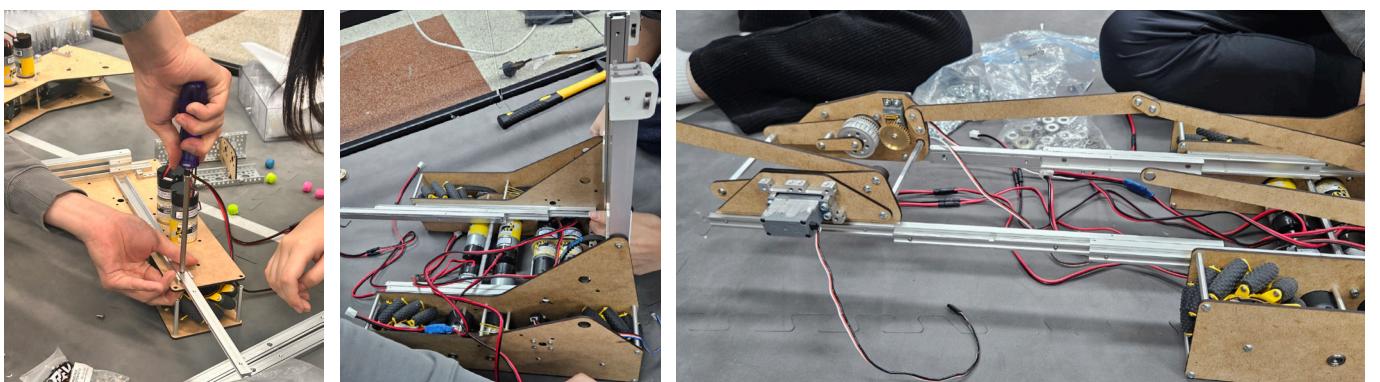
2. 수직 리니어 조립 및 설치



3d 프린터로 출력한 리니어 가이드와 풀리 마운트를 차례대로 고정하여 3단 리니어를 조립함. 10mm M4 접시머리 나사를 이용하여 인접한 리니어와 리니어의 슬라이드 부분을 결착하였고 풀리는 따로 고정하지 않았음.

본래의 차체 위에 수직 리니어를 부착함.

3. 수평 리니어 제작 및 설치



차체가 샘플을 집는 용도로 수평 리니어도 제작함. 이는 수직 리니어와 같은 방법으로 조립했으며 차체의 안쪽에 수평하게 부착함. 수평 리니어는 4단을 사용하였으며, 팔을 부착할 계획을 가지고 있었음.

수평 리니어를 부착한 이후에는 수평 리니어를 접었다가 펼 수 있도록 팔을 설치함. 리니어의 끝과 끝을 하나의 관절을 사이에 둔 2개의 긴 막대를 활용하여 연결했고, 차체쪽 끝의 막대에는 헥스모터를 연결하여 모터가 제공하는 토크로 수평 리니어를 접었다가 펼 수 있도록 만듦. 결과물은 오른쪽 사진의 모습.

2025년 01월 12일 (일)

구분 | WINTER_EN_BUILDING

책임자 | 23 김준성



작성/편집 | 23 김진용, 23 류승완

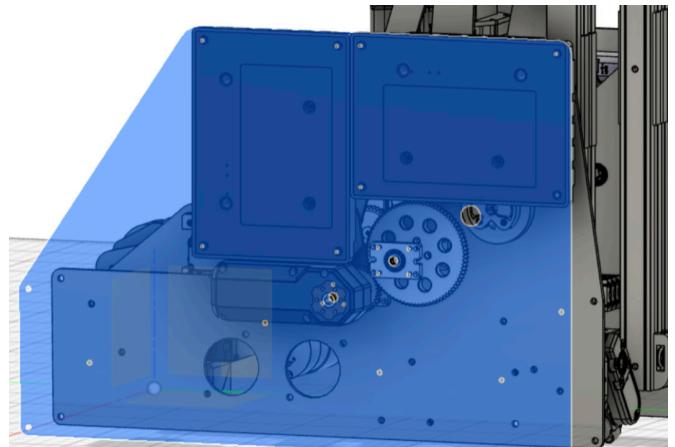
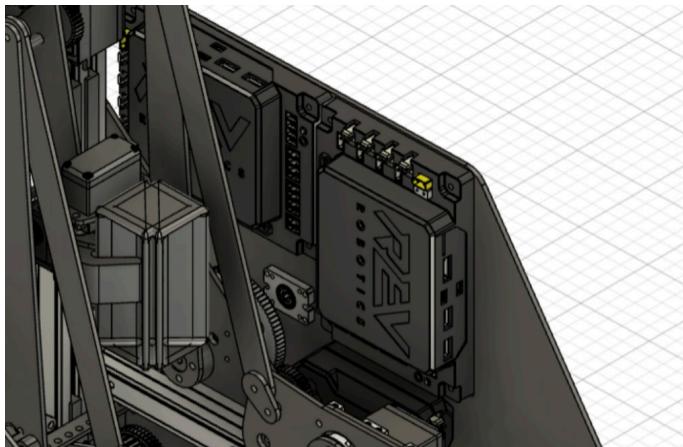
작업 참여 | 전체

1. 설계 및 차체 골격 수정

리니어를 구동시키는 기어를 수정한 새로운 모델을 cad로 설계하여 기존에 제작한 차체를 분해하고 수정하여 재조립하는 과정을 거침. 재조립하는 과정은 이번대회가 처음인 빌더들이 차제의 구조와 조립과정을 배우게 하는데에도 의의가 있었음. 재조립하는 과정에서 몇가지 주의사항을 깨달을 수 있었음.

1) 리니어 전개 방식 수정

앞서 제작한 차체에서, 굽은 팔을 모터로 회전시켜 수평 리니어를 전개하는 방식을 사용하였는데, 이 때 모터의 토크가 부족하여 전개가 제대로 이루어지지 않음. 이에 따라 기어비를 조정하여 토크를 크게 할 수 있도록 설계를 수정함. 또, 모터 조작을 위한 허브를 장착할 공간이 없어 허브를 장착할 수 있는 공간을 추가하여 전체 차체 골격을 수정함. 수정된 차체 골격을 레이저커터를 통해 MDF로 프린트함.



2) 수평 리니어 나사 걸림 문제

Horizontal Linear 결합시 나사가 서로 걸리는 문제점이 발견됨. 리니어에 평소에는 수평 방향으로만 힘이 가해지기에 큰 문제는 없었으나, 수직 방향으로 힘을 가하면 나사에 걸려 확장에 방해가 된다는 것을 확인하였음.

이는 첫번째 나사를 타이트하게 조여야 리니어가 걸리지 않고 부드럽게 넘어갔음.



3) MDF 강도 문제

Stand off를 먼저 조립을 하고 조립을 하는데 stand off가 조립되어있는 나사구멍과 가장자리의 간격이 작아 MDF가 파손되는 일이 빈번하게 일어남. 이를 위해 최종 prototype에서는 MDF보다는 강화된 재료를 사용하거나 가장자리의 나사 구멍을 조금 여유있게 설계할 필요성을 느낀



4) HEX 모터 나사 구멍 위치

설계상에서 Hex Motor를 설치하였을 때 맞은 편 stand off의 나사구멍이 막히는 문제점을 발견하게 됨. 재조립과정에서는 stand off를 한쪽만 고정하여 문제를 해결함. 이는 추후 Hex Motor의 위치를 조금 이동시켜 해결을 할 예정임.



5) 베어링 구멍.

CAD파일상에서는 motor의 베어링 구멍이 두개를 설계되어 있었으나, 레이저 커팅 과정에서 문제가 발생하여 구멍이 커팅 되지 않았음.

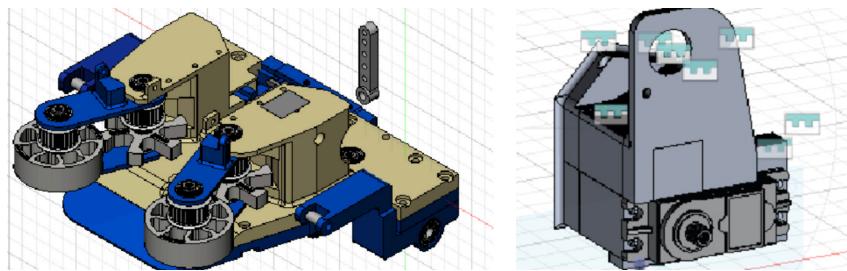


작성/편집 | 24 함주원
작업 참여 | 24 함주원

2. 개선된 롤링 Eater 설계

1월 9일에 롤링 이터 제작을 진행하며 조금 더 다양한 각도의 샘플을 원활히 먹기 위해 새로운 형태의 이터를 제작함. Bluebot과 미팅을 통해 전달 받은 CAD 파일을 기반으로 설계하는 방안으로 결정함.

Bluebot이 사용한 부품을 기반으로 설계를 진행하려고 했으나, DC모터를 Superspeed servo 2개로 대체하고 REV color sensor의 위치를 수정하는 등, 일부 역설계를 시작함. 특히 3D 프린터로 출력하기 위해 돌기와 홈을 이용해 조립하는 형식으로 수정 방향을 설정함.



다음과 같이 서보를 위해 새로 마운트를 설계하였고, 타이밍 벨트를 이용해 동력을 전달할 수 있도록 수정함. 샘플의 이탈을 방지하기 위해 이터의 뒷면에 Torque servo를 닫. 컬러센서가 색을 인식하여 색이 다르다면 배출하고 원하는 색상이면 갖고 있도록 함. 공간 상 컬러센서의 위치를 뒤쪽으로 옮김. 판과 판 사이를 고정하기 위해 축을 프린트하여 고정하는 방식을 채택함. 나머지 부품 역시 이미 보유한 부품을 활용하여 제작할 수 있도록 설계를 수정함.

2025년 01월 13일 (월)

구분 | WINTER_EN_BUILDING

책임자 | 22 김수기



작성/편집 | 22 최성빈

작업 참여 | 22 최성빈, 22 강현빈

1. 번호판 제작

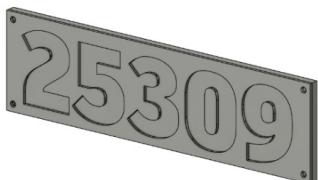
ROBOT SIGN 관련 규정을 참고하여 일단 가장 간단한 직사각형 판에 가로로 팀 번호를 붙여 놓은 형태로 제작에 들어감. 팀 번호판의 숫자 부분을 양각(Emboss)이나 음각(Engrave)으로 표현할지, 아니면 종이나 코팅지를 붙이거나 마스킹 처리를 해서 숫자 부분의 색만 다르게 도색하는 방법 등 다양한 방식을 고민해 봄.

양각으로 표현할 경우 숫자 부분의 아크릴 조각들을 글루건이나 아크릴 접착제를 이용하여 고정시켜야 하므로 추후 내구성 문제가 발생할 수 있다고 생각하였고 이것을 해결하고자 볼트를 사용하게 되면 볼트 구멍을 뚫어야 하므로 미관 상 좋지 않다고 판단하였음.

음각으로 표현할 경우에는 마스킹 작업 없이 숫자 부분을 다른 색으로 표시할 수 있고 도색 작업도 숫자와 바탕 부분으로 나누어 간단히 처리할 수 있음.

시트지에 팀 로고와 번호를 프린트한 후 아크릴에 붙이는 방법은 결과물을 깔끔하겠지만 작업이 번거롭고 시트지 인쇄를 따로 외주를 맡겨야 함. 마스킹 처리를 하는 방법은 몇 번의 시행착오를 겪을 것이라고 생각하지만 결과물의 두께가 한 장으로 얇고 작업이 순조롭게 된다면 좋은 결과를 기대할 수 있다고 생각함. 이중에서 음각과 마스킹을 사용하는 방법을 시도해보기로 결정함.

1) 음각



작년에 아크릴에 래커를 이용하여 도색하였을 때 외관상 문제가 없었던 것으로 기억함. 그래서 따로 서페이서를 뿌리지 않고 아크릴의 보호 필름을 제거한 후 바로 래커로 도색함. 그 결과는 아래 사진과 같이 도막이 충분히 두껍지 않아 도포량에 따라 아크릴 뒤의 색이 비춰보였음. 그래서 그냥 서페이서를 뿌리고 도색하기로 결정함. 작은 크기의 조각들은 래커를 뿌릴 때의 힘 때문에 밀려 도색이 어려웠으므로 몇 번의 시행착오 끝에, 도색할 조각들을 테이프로 고정하여 도색하는 방법을 이용함.



이전에 결정한 내용대로 서페이서를 먼저 아크릴 번호판 표면에 도포하고 충분히 건조한 다음, 락카를 이용하여 빨간색, 파란색으로 도색함. 서페이서 없이 도색했을 때보다 도막이 두꺼워져 뒤가 비치는 문제는 해결되었으나 오히려 표면에 도료가 부분부분 뭉쳐 거칠게 변함. 하지만 색이 섞이지 않는 것이 더 중요하므로 서페이서를 뿌리는 편이 낫다고 생각함. 서페이서를 뿌리고 1200방 사포로 표면을 살짝 정리해주는 것이 좋을 것 같음.

2) 마스킹

마스킹 테이프를 일일이 재단하여 도색하는 것은 번거롭고 시간도 오래 걸리므로 레이저 커터를 활용하는 방법을 시도해봄. RDWorks에서 숫자 부분의 외곽선을 Cut 레이어로 설정하고 Speed를 50mm/s, Power를 20%로 설정하여 작업함. 화재가 난다거나 하는 등의 안전 사고는 발생하지 않았고 결과물은 깔끔하게 잘 나왔음.



그런데 흰색 래커로 도색하고 마스킹 테이프를 제거해보니 생각만큼 작업물의 질이 좋지 않았음. 빨간색과 파란색 래커 도색을 옆에서 먼저 하고 그다음에 흰색으로 도색을 했더니 착색되어 얼룩이 생겼고, 레이저 커팅으로 파인 부분이 백화되는 등 약간의 문제가 생김. 그래서 레이저로 인해 파이는 깊이를 최소화하기 위해 레이저 커터의 Speed와 Power를 조금씩 바꾸면서 테스트해 봄. Speed를 4배, Power를 1/2배인 200mm/s, 10%로 설정해서 레이저 커터를 가동해 보았지만 파이는 깊이가 같았음. 이와 같은 결과로부터 레이저 커터 소프트웨어 상에 속력 제한이 걸려있다는 사실을 알 수 있었음.



도색 작업 과정!

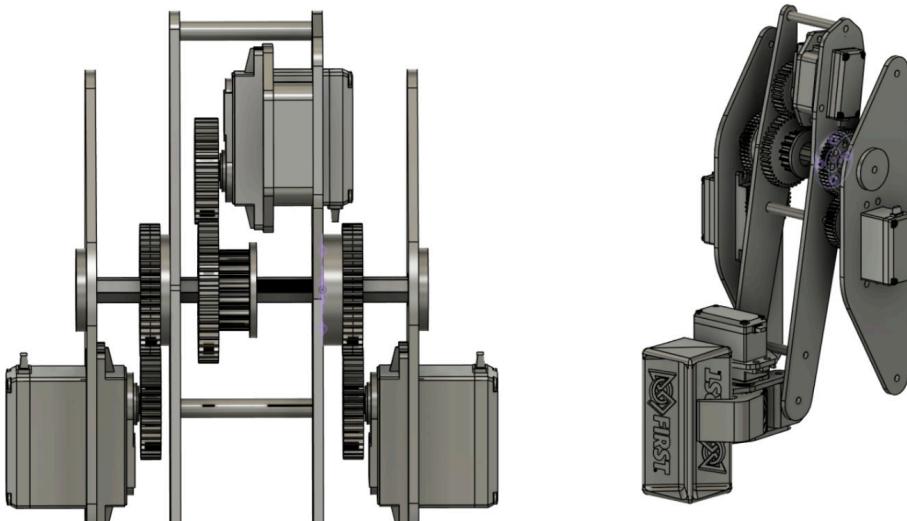
1. 레이저 커터로 5T 아크릴 판재를 번호판 크기의 직사각형으로 절단 및 레이저 커터 위치 고정.
2. 절단한 직사각형의 아크릴 조각 한면의 보호 필름을 제거하고 서페이서 도포.
3. 서페이서가 충분히 건조된 후에 고운 사포로 표면을 살짝 정리.
4. 마스킹 테이프를 아크릴 조각 크기에 맞추어 꼼꼼히 붙여줌.
5. 1번에서 레이저 커터로 절단한 부분에 맞추어 아크릴 조각을 배치하고 레이저 커터로 번호 부분을 절단(Speed 100mm/s, Power 10%로 설정).
6. 마스킹 테이프의 이음새를 글루건으로 고정한 후, 번호 부분(면적이 작은 조각들)을 먼저 제거하고 락카로 도색(제거한 마스킹 테이프 조각들은 따로 보관).
7. 번호 부분에 수성 바니쉬를 발라주고 열풍기로 건조(5분이면 마름).
8. 번호 부분 마스킹 조각들을 다시 붙이고 배경 부분의 마스킹을 제거, 그리고 배경 부분을 락카로 도색.
9. 배경 부분도 수성 바니쉬를 바르고 열풍기로 건조.
10. 작업 마무리.



작성/편집 | 24 이소민
작업 참여 | 전체

2. 수평 리니어 팔 제작

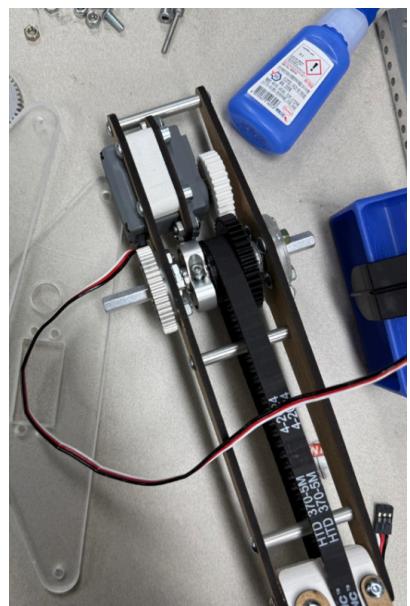
설계를 토대로 수평 리니어에 부착될 팔 제작을 시작함. 서모 마운트와 기본적인 판은 MDF로 제작하였고, 기어의 위치를 고정하기 위한 스페이서를 아크릴로 제작하였음.



안쪽 서보모터에 부착되는 검정색 기어는 3D 프린터로 출력하고 나머지 스퍼기어들은 메탈기어를 사용함.

본체에 부착되는 양쪽의 서보모터에 달린 기어가 40 tooth 기어인데 2개가 필요한 40 tooth 기어가 1개밖에 남지않아, 30 tooth 기어로 설계를 수정함. 서보모터 마운트에서 서보모터 구멍과 볼트 구멍의 위치를 +y 방향으로 약 4mm 이동하여 수정함.

또한 회전축이 본체에 고정이 되어야 하는데, 기존 방법은 와셔 모양의 MDF를 순간접착제로 붙여 고정하는 방법이었음. 하지만 접착제를 사용하는 설계의 단점이 너무 많아, 본체에 축이 연결되는 부분의 구멍을 축에 맞는 hex 모양으로 수정하여 고정함.



모두 제작한 다음, 팔 전체의 회전을 담당하는 기어 부분이 서보모터의 위치 문제로 180도 이상 회전하지 못하는 문제점이 발생함. 이를 해결하기 위해 축의 길이를 늘리고, 스페이서를 추가하여 팔 회전부가 360도 회전할 수 있게 설계를 수정함.

2025년 01월 14일 (화)

구분 | WINTER_EN_BUILDING

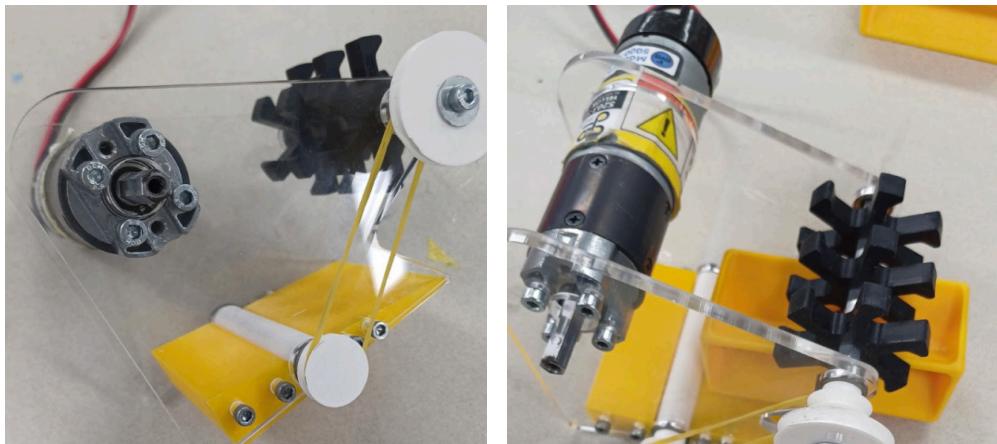
책임자 | 24 이소민



작성/편집 | 24 이소민
작업 참여 | 24 이소민

1. 롤링 Eater 완성

기존에 제작을 시도했던 롤링 이터를 총 6회정도의 시도 끝에 완성함.



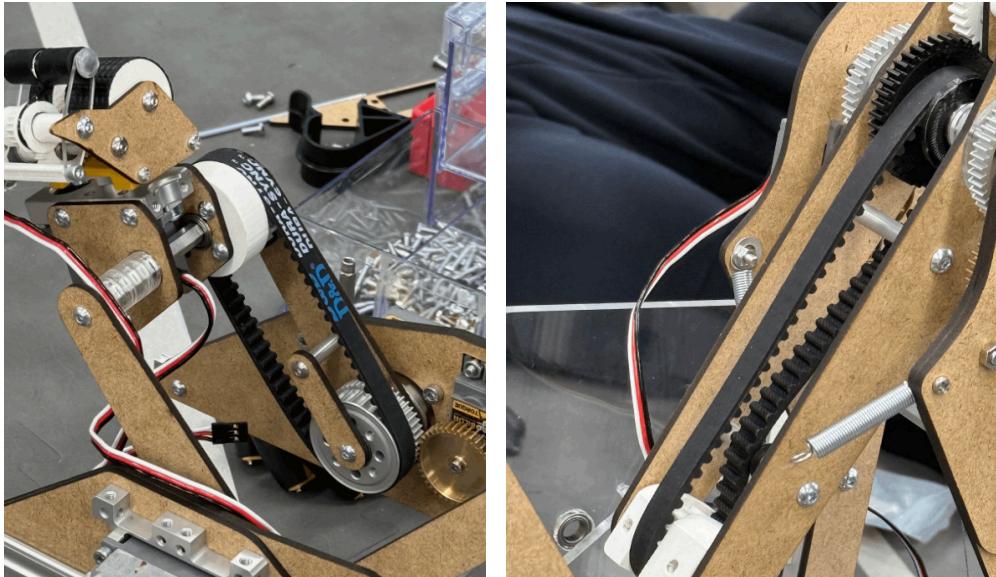
앞서 말했던, 둘러 높이에 의한 문제와, 고무 튜브의 탄성력 부족등을 보완한 모델을 설계하여 specimen을 완벽하게 intake 할 수 있는 모델을 설계했고 출력하여 완성함. 양옆의 판은 5T 아크릴로 제작하였고, 나머지 부품들은 3D 프린터로 출력함.



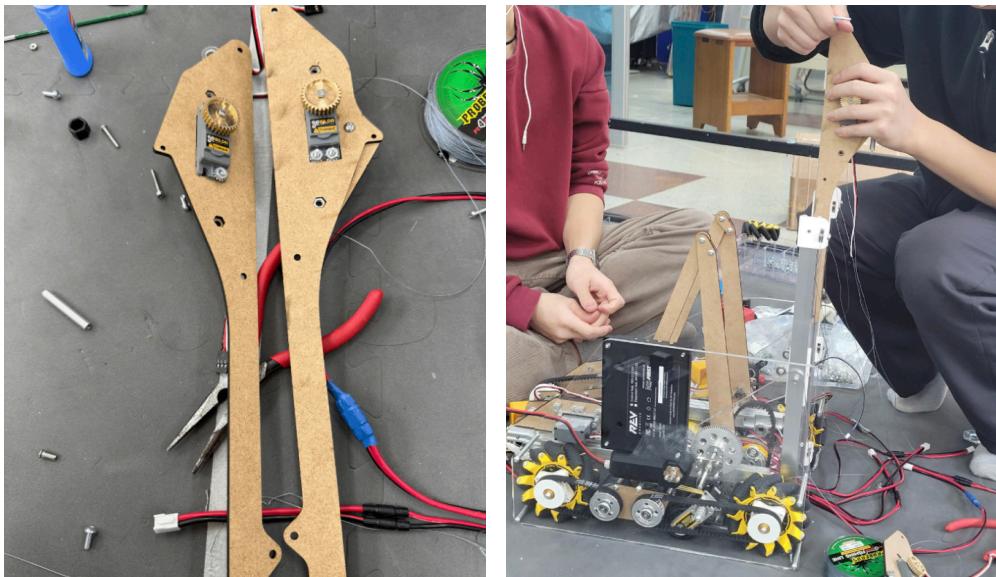
작성/편집 | 22 장호원
작업 참여 | 전체

2. 팔 설계 수정

전날 팔에 있는 서보 모터와 기어가 걸려서 팔이 360도 돌아가지 않는 문제를 서보와 기어의 위치를 조정하여 설계한 것을 제작함. 설계를 수정한 결과 팔이 360도 원활하게 잘 돌아가는 것을 확인함.

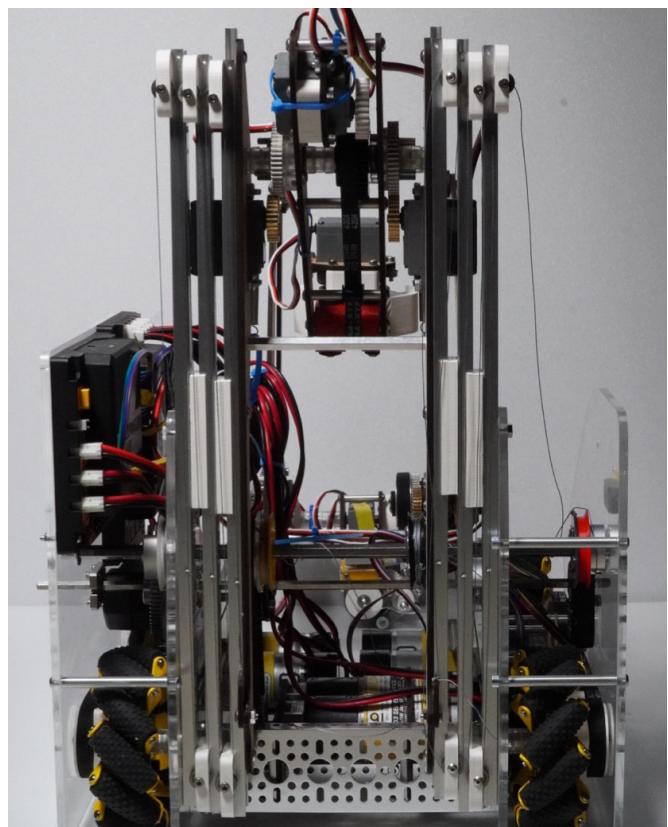
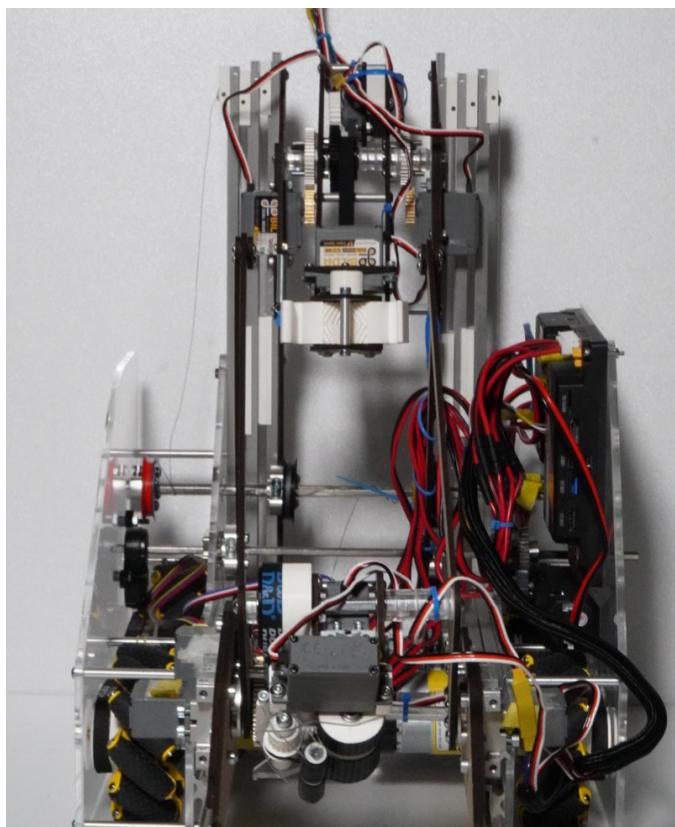
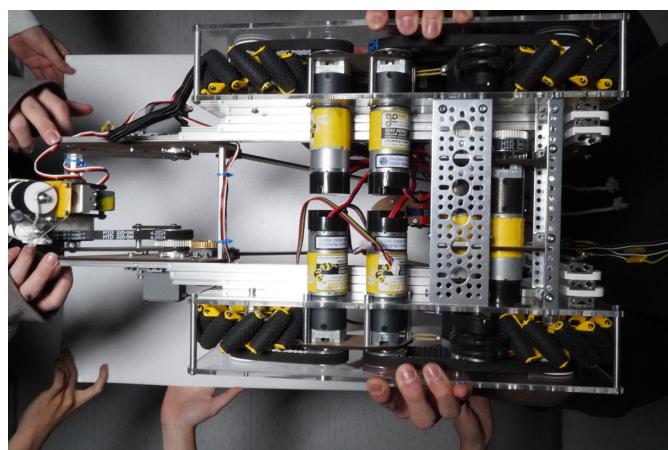
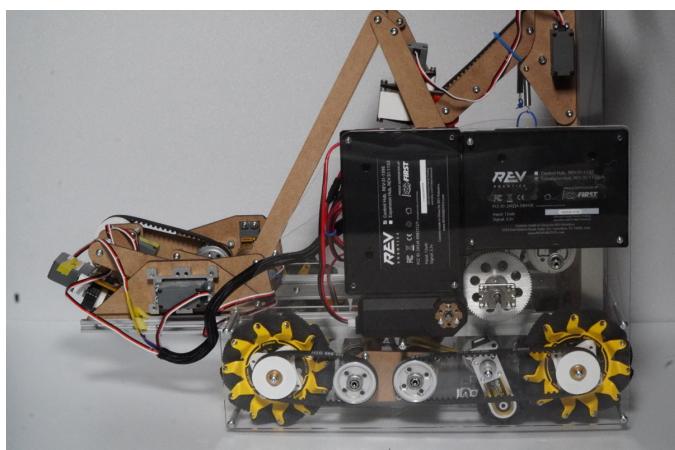
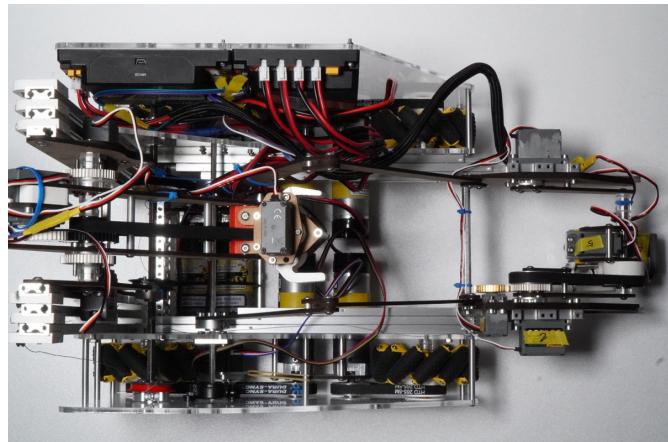
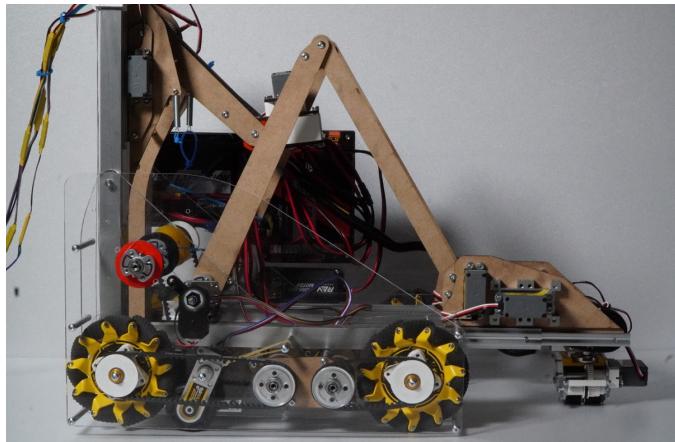


팔 뿐만 아니라 설계에 맞게 수직 리니어쪽 마운트도 수정하여 다시 수직리니어와 팔을 부착하여 연결함. 이외에도 실제 작동시켜보는 과정에서 생긴 마이너한 문제들을 해결함.



이렇게 최종적으로 다시 조립한 후 구동시켜본 결과 잘 작동하는 것으로 확인되었으며, 마지막으로 선 정리를 진행함.

이로써 로봇을 완성함. 완성한 로봇은 이터, 수평 리니어, 수직 리니어, 팔이 유기적으로 연결되어 견본을 옮기는 가장 기본적인 움직임을 비롯해 *Into the Deep*의 미션을 수행하기에 적합하며 차체 또한 메카님 휠과 오도메트리로 자유롭게 이동이 가능함. 직접 경기장에서 경기 시뮬레이션을 해본 결과 미션을 문제 없이 효율적으로 잘 수행하는 것으로 확인됨.



ENGINEERING NOTE
_PROGRAMMING
프로그래밍

해당 섹션은 OnBot Java를 사용해 로봇을 프로그래밍한
과정과 결과에 대해 다룹니다.

목차 |

1. 프로그래밍 방식 개요
2. 프로그래밍 과정 및 시행착오

I. 사전 준비

저번 대회보다 더욱 효율적이고 강력한 기능을 가진 로봇 컨트롤러를 제작하기 위하여 개발 환경, 라이브러리, 개발 방식 등 다양한 방면에서 새로운 기술을 학습하고 연구하는 과정을 거침. 공부한 내용을 바탕으로 기존보다 더욱 발전된 코드를 작성하여 로봇의 구동, 특히 자율 주행 측면에서 큰 이점을 얻을 수 있었음.

1. Android Studio 기반의 개발 환경 구축

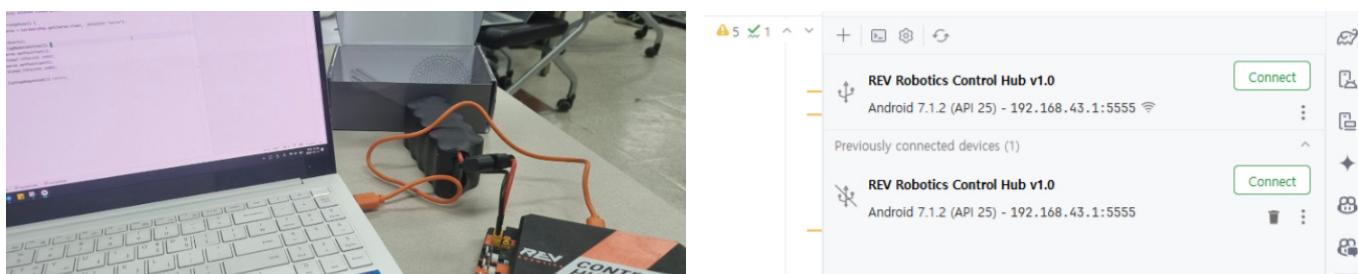


Rev Client에서 빌드했을 때의 단점!

-] 코드를 전부 읽기는 과정이 **번거로우며**, 그 과정에서 실수가 발생할 수 있음.
-] 여러 사람이 함께 Control Hub에 연결하여 코드를 빌드할 수 없음.
-] **상수값 수정 및 버그 수정 등은 코드를 다시 읽기는 작업을 최소화 하기 위해 Rev Hardware Client에서 직접 코드를 수정할 수 있음.** 하지만 이 경우 Android Studio로 **다시 코드를 읽겨야 하는 불편함**이 있으며, 이를 잊어먹는다면 수정한 값이 날아가 추후에 **불필요한 노동**을 수반하며, 심지어는 더 큰 버그로 발전하는 경우가 있음.

이러한 단점을 극복하기 위하여 이번 시즌부터는 Android Studio에서 직접 Robot Controller를 빌드하는 방법을 찾아내었음.

Android Studio에서 Control Hub를 디바이스로 인식하면 Control Hub에 Robot Controller를 빌드할 수 있음. Android Studio에 Control Hub를 연결하기 위해서는 연결선을 이용해서 직접 연결하는 방법과 Control Hub WiFi 연결과 Android Studio의 ADB Wi-Fi 플러그인을 통한 무선 연결이 가능함. 연결선을 활용한 방법은 인터넷을 사용하면서 동시에 빌드가 가능하다는 점에서 공부를 병행하며 테스트를 할 때 활용하기 적절한 방법임. 반대로 무선 연결은 원거리에서도 쉽게 Robot Controller를 빌드할 수 있다는 점에서 로봇을 제작한 후 그에 맞는 코드를 작성해 빌드할 때 유용하게 쓰일 수 있음.



Rev Hardware Client는 Driver Hub 및 Control Hub의 앱 및 펌웨어 버전 관리, 로그 확인, 그리고 WiFi 설정 관리를 위해서만 사용하고, 그 외의 소스코드 빌드 작업은 Android Studio로 바로 빌드하는 이러한 시스템은 기존의 단점을 보완하고, Android Studio의 검증된 최적화를 사용할 수 있으며, Github를 통해 공유되는 코드 스페이스를 직접적으로 활용하여 동시에 소스코드 빌드가 가능하다는 점에서 기존에 방식에 비해 큰 이점이 있으리라 기대됨.

2. Road Runner

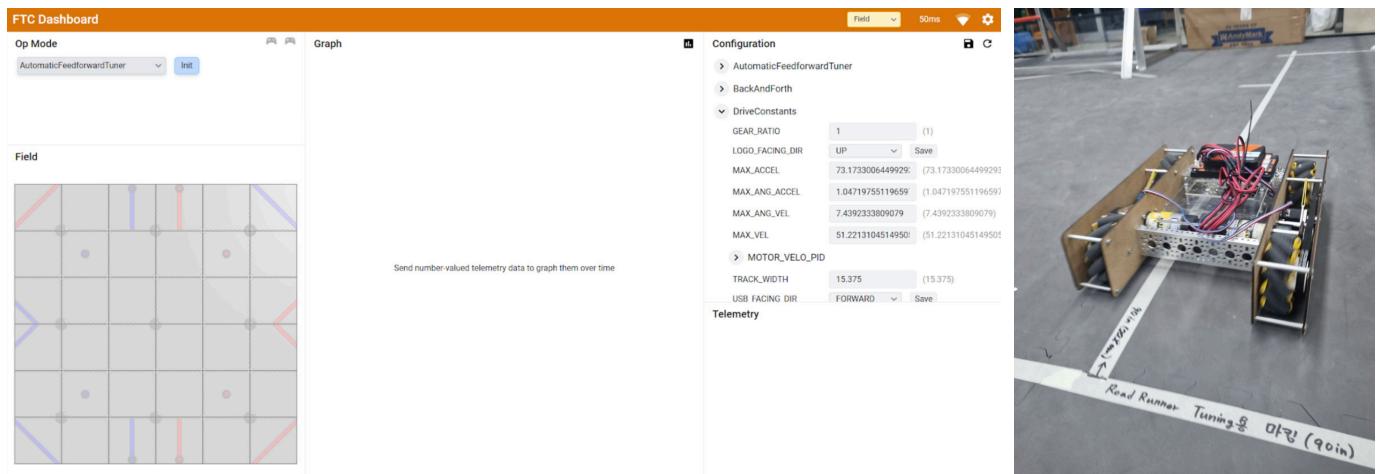
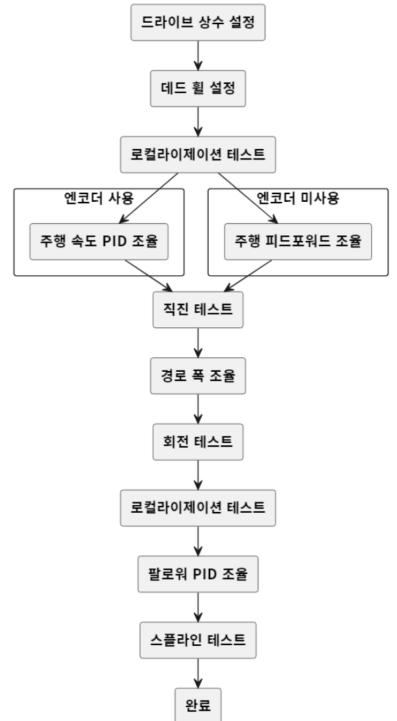
Road Runner는 자율 주행시에 오도메트리와 IMU 센서를 활용한 자율 주행 기능을 편리하게 제공하기 위해 ACME Robotics 등이 제작한 모션 플래닝 라이브러리임. 대부분의 해외팀은 정확한 Localization과 자율 주행을 위해 Road Runner를 사용하며, 이를 통해 AutoOp에서 높은 점수를 얻을 수 있음. 저번 대회에서는 IMU와 로봇 바퀴 엔코더에 기반한 자율 주행 코드를 작성했기에 정확도가 상당히 떨어져 추가적인 점수 없이 Randomization만을 목표로 하는 간단한 자율 주행만 구현할 수 있었음. 세계 대회 출전을 기점으로 오도메트리를 구매하였으나, 당시에는 시간적인 한계로 인하여 Road Runner에 대한 공부 없이 자체적인 Localization 코드를 작성하였으나, PID를 사용하지 않아 단계별 정확한 정렬을 위해 많은 시간이 소요되었고, 정확한 튜닝이 불가능하여 정밀도의 한계가 명확하였음.

저번 대회의 한계를 극복하고, 정확한 Localization을 통해 AutoOp의 효율을 극대화하기 위하여 Road Runner를 공부함. 또한, 추가로 한국의 FTC 팀들이 Road Runner를 쉽게 접하고 공부할 수 있도록 한국어 기반 Road Runner 가이드 웹사이트를 제작함 (Outreach 섹션 참고).

1) 튜닝 (Tuning)

Road Runner의 튜닝을 시도하였음. ACME Robotics의 Road Runner Quickstart 레포지토리를 가져와 개발 환경을 구축하고, 절차에 맞게 튜닝 과정을 수행함. 튜닝 절차는 오른쪽 그림과 같다. 튜닝 테스트를 통해 필요한 공간의 크기, Localization의 성능을 극대화하기 위한 오도메트리의 위치 설정 등 로봇 제작 시 고려해야 할 점, 그래프와 로봇의 움직임 해석한 후 이에 기반한 PID K값 수정 방법 등을 터득하였으며, 이는 추후 로봇을 제작하여 자율 주행 시스템을 구축하는데에 큰 도움이 되리라 생각됨.

튜닝 결과 상당히 정확하고 빠르게 원하는 위치로 이동하고 정렬할 수 있음을 확인하였음. 하지만 첫 튜닝인만큼 각 과정마다 약간의 오류가 있었고, 주행이 여러번 반복되면 될수록 오차가 커지는 것이 발견되었음. 그럼에도 불구하고 자율 주행이 정확하게 되는 것을 통해 각 과정에서 튜닝을 확실하게 하고 넘어간다면 실제 로봇에서의 자율 주행 성능이 크게 향상될 수 있으리라 결론 지을 수 있음.



2) 경로 설정 (Trajectory)

경로를 설정하고 Road Runner가 작동하는 모습을 직접적으로 확인하였음. 정확한 위치와 방향으로의 움직임, 직선적인 움직임 뿐만 아니라 곡선적인 움직임도 가능한 시스템, 움직임 중간에 마커를 추가하여 다른 장치를 움직이는 등 추가 장치를 조작할 수 있는 기능 등 다양한 기능을 테스트 해봄으로써 Road Runner의 가능성을 확인해보고, 이를 기반으로 자율 주행 코드를 어떻게 구성할 수 있을지를 생각해봄. 움직임이 매우 정확하고 빠르며, 마커 등 강력한 기능을 적절히 사용한다면 높은 성능의 자율 주행 시스템을 구현할 수 있으리라 생각됨. 특히 기존의 코드 구조는 비동기적인 처리를 위해 오버헤드가 많이 발생하는 시스템인지라 성능 저하가 많이 발생하였으나, Road Runner의 시스템을 적극 활용한다면 AutoOp는 물론 TeleOp 시기에도 높은 효율을 낼 수 있을 것으로 기대됨.



RoadRunner 테스트에 사용한 차체



RoadRunner 테스트는 프로토타입으로 제작한 차체를 사용했습니다. 자세한 내용은 학기 중 작성한 Building EN에 기록되어 있습니다.

실제로 사용할 Drivetrain과 같은 구조를 차용하였으며, 보정 등 다양한 테스트를 진행했습니다.

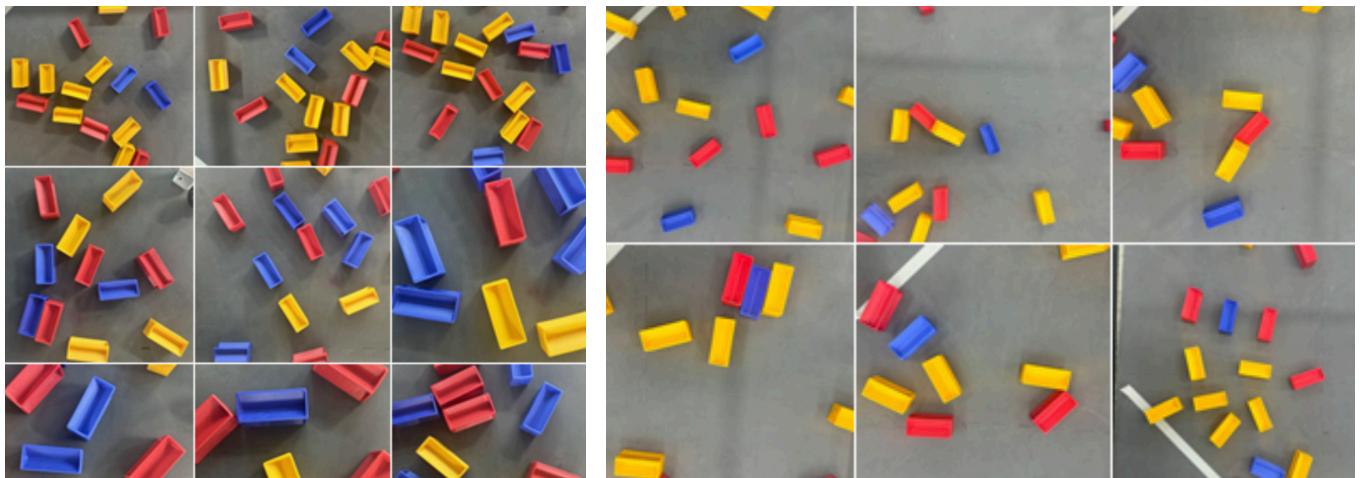
3. Vision

샘플을 정확하게 인식하고, 위치와 색깔 정보를 활용하여 자율 주행과 원활한 드라이버의 로봇 조종을 위하여 웹캠을 활용한 비전을 공부하고 연구하였음.

1) 머신 러닝

실시간 웹캠 피드에서 블록을 정확하게 인식하고 구분하기 위해 YOLO (You Only Look Once) 모델과 Roboflow를 활용함. Python을 이용해 이러한 시스템을 구현함으로써 실시간 객체 인식의 가능성을 탐구함.

YOLO 모델을 효과적으로 사용하기 위해서는 충분한 학습 데이터가 필요하기에, 세가지 색상의 블록을 다양한 각도와 조명 조건에서 촬영된 블록 이미지들을 수집함.



YOLO 모델이란?

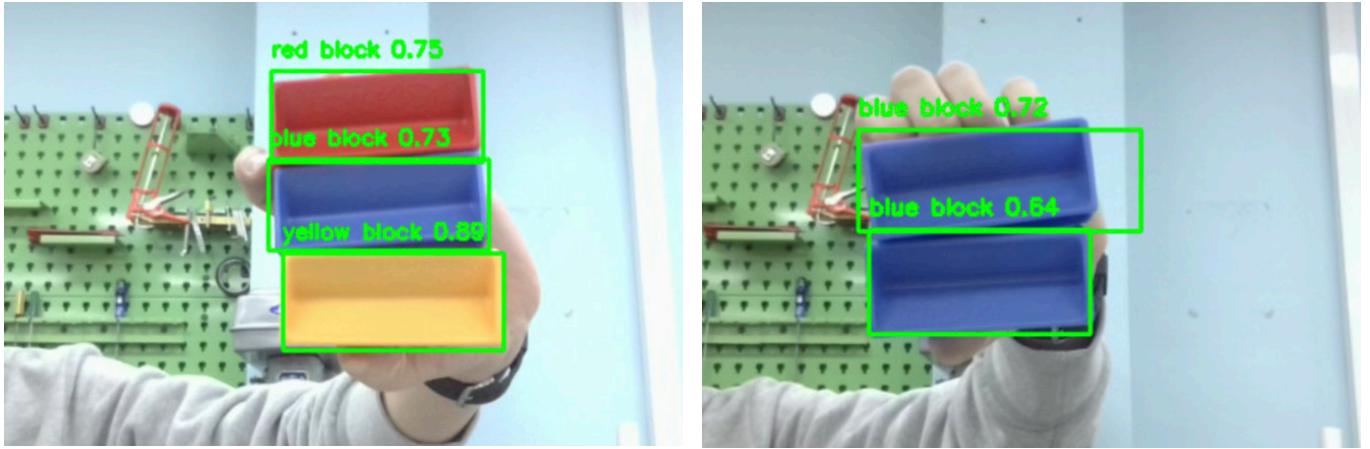


YOLO는 객체 인식 분야에서 널리 사용되는 실시간 객체 탐지 모델입니다. YOLO의 주요 강점은 이미지를 한 번에 처리하여 여러 객체를 동시에 인식할 수 있다는 점입니다.

기존의 객체 인식 모델 | 이미지를 여러 번 분석

YOLO | 단일 신경망을 통해 전체 이미지를 한 번에 처리

수집된 데이터는 Roboflow를 통해 전처리되었으며, 이미지에 바운딩 박스를 그리고 클래스 라벨을 지정하는 주석 작업을 거쳤음. 이러한 과정을 통해 모델이 블록 객체를 다양한 상황에서 정확하게 인식할 수 있도록 데이터셋을 준비하였음. YOLOv8 모델을 선택하여 Roboflow에서 제공하는 도구를 사용해 학습을 진행하였으며, 하이퍼파라미터 튜닝을 통해 모델의 성능을 최적화함. 학습된 모델은 높은 정확도로 블록 객체를 인식할 수 있었음.



Roboflow는 컴퓨터 비전 프로젝트를 위한 데이터 관리 및 전처리 도구로, 본 프로젝트에서 핵심적인 역할을 하였음. Roboflow를 통해 이미지 데이터셋을 효율적으로 관리하고, 주석 작업을 간편하게 수행할 수 있었음. 데이터 증강 기능을 활용하여 데이터셋의 다양성을 높였으며, 전처리 과정을 자동화함으로써 모델 학습에 최적화된 데이터를 준비할 수 있었음. 또한, Roboflow의 API를 통해 학습된 모델을 손쉽게 배포하고, Python과 연동하여 실시간 객체 인식 시스템을 구축하는 데 기여하였음.

2) 두 번째 모델 제작 및 테스트

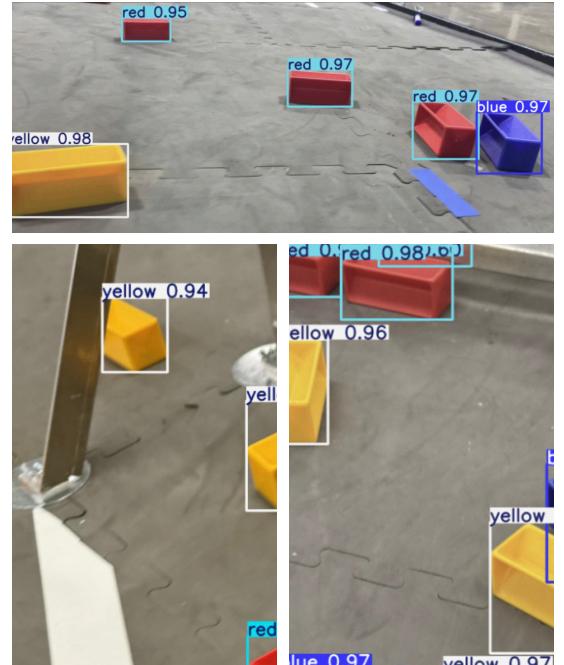
다음으로는 YOLO v8보다 더 높은 성능의 최신 버전인 YOLO v11 모델과 Roboflow에 올라와있는 오픈 데이터셋을 활용하여 심층 학습을 진행하였음. Roboflow의 데이터 증강 기법을 활용하여 기존의 데이터셋을 회전, 확대, 축소 등의 변형을 거쳤으며, 색조, 채도, 밝기 등에 다양성을 추가하여 다양한 환경에서의 데이터를 학습할 수 있도록 데이터셋을 구축함. 이를 통해 총 203개의 이미지를 487개의 이미지로 증폭시켜 학습에 활용하였음.

학습은 4번에 걸쳐서 진행하였음. 각 학습에서 사용한 데이터셋 원본은 동일하지만, 데이터 증강 기법을 활용해 서로 다른 세 가지의 데이터셋을 제작함. 학습 결과는 아래와 같음.

Version	mAP	Precision	Recall
Ver 1	99.2%	95.4%	99.1%
Ver 2	99.2%	95.1%	100.0%
Ver 3	99.2%	94.3%	99.8%
Ver 4	97.5%	95.1%	91.3%

학습 결과 2번째 버전에서 모든 지표의 값이 가장 높게 확인되었음. 학습이 거듭되며 2번째 버전까지는 학습 결과 발전이 있었지만, 그 이후로는 오버피팅에 따라 정확도가 감소한 것으로 추정됨. 아래 그림은 2번째 버전 YOLO 모델의 객체 감지 모습이며, 이와 같이 매우 높은 정확도로 샘플의 종류와 위치를 특정할 수 있는 것이 확인됨.

YOLO 모델을 학습시키면 PyTorch 가중치 파일을 얻을 수 있고, 이를 FTC SDK에서 이용하기 위해서는 TFlite 파일로 변환시켜야 함. 파일을 변환시키는 건 손쉽게 가능했지만, Output Layer 구조가 달라서 FTC SDK에서 이를 활용할 수 없었으며, 찾아본 결과 FTC SDK에서 지원하는 TensorFlow의 버전이 낮아서 호환이 안 되는 것이 문제였음. 결론적으로 인공지능 기술이 발전함에 따라 TensorFlow가 빠르게 변화함에도 불구하고 FTC에서 이를 즉각적으로 반영하는 것이 어려워 이번 시즌을 기점으로 TensorFlow를 활용한 객체 감지 기능의 공식적인 지원이 중단되었고, 이에 따라 우리 팀은 Limelight 또는 HuskeyLens 등 다른 하드웨어를 활용한 머신러닝 또는 Open CV의 이미지 처리를 통한 객체 감지를 수행해야 했음.



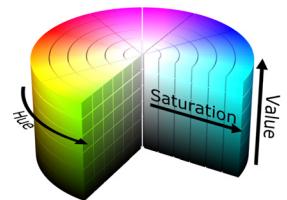
3) 결론

결론적으로 머신 러닝 테스트 통해 YOLO와 Roboflow를 활용한 실시간 객체 감지 모델을 생성하는 것이 가능하며, 충분히 쓸 수 있을 정도의 높은 정확도를 가짐을 확인하였음. YOLO의 빠른 객체 인식 능력과 Roboflow의 효율적인 데이터 관리 및 전처리 기능을 결합함으로써 높은 정확도와 실시간 처리를 동시에 달성할 수 있었음. 최종적으로, 실시간 객체 인식 기술의 실용성과 가능성을 확인하였으며, 이를 기반으로 더욱 발전된 응용 시스템을 개발할 수 있는 기반을 마련하였음. 물론 이번 시즌부터 FTC가 TensorFlow 기반 인공지능 모델 지원을 공식적으로 종료함에 따라 현재는 당장 사용할 수 없는 기술이지만, 이후에 Limelight 또는 HuskeyLens 등 인공지능 카메라를 구매한다면 이 때의 경험을 바탕으로 높은 수준의 객체 인식 시스템을 구성하여 자율 주행에 활용할 수 있을 것으로 기대됨.

4. OpenCV

머신러닝 외에도 머신러닝에 비해 빠르고 가벼운 이미지 처리를 통해 물체를 감지하는 방법을 연구하였음. 이를 위하여 이미지 처리 라이브러리인 OpenCV를 사용함. 기본적인 아이디어는 샘플과 바닥 사이의 급격한 색상차를 활용해 감지하는 것으로, 색의 특성을 잘 반영하는 색상 표현 방식인 HSV 형식을 사용하여 바닥과 샘플을 분리함.

샘플은 경기장 바닥에 비해 상대적으로 채도가 높다는 특성을 활용해 채도(Saturation, S Value) 값을 중심으로 바닥과 샘플을 분리하였으며, 추가적으로 다양한 조명 환경에서 샘플이 가질 수 있는 최소 명도(Value, V Value)를 실험적으로 구하여 더욱 정밀한 샘플 탐지를 구현함. 샘플 사이의 색상 구분은 색조(Hue, H Value)를 사용하였으며, 그 범위를 넉넉하게 하여 샘플 감지에 방해가 되지 않도록 처리함.



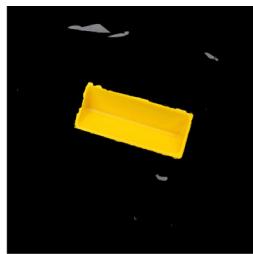
알고리즘 작동 방식



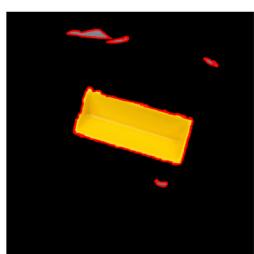
1. 빨간색, 파란색, 노란색 샘플 각각에 대하여 HSV 범위 지정



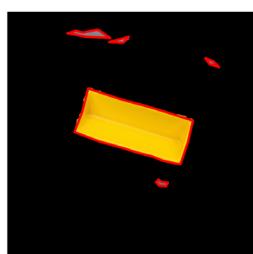
2. 이미지 마스킹을 활용하여 범위에 해당하는 부분 추출



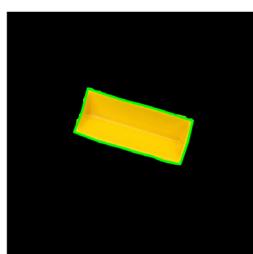
3. 그룹화된 마스크에서 노이즈 제거



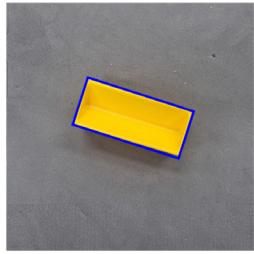
4. 필터링된 마스크에서 모서리 부분 추출



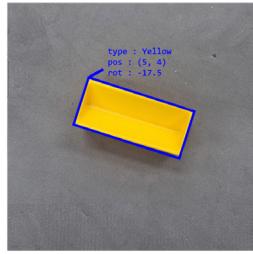
5. 모서리 형태 단순화



6. 윤곽의 넓이 계산 후 최소 기준을 충족하지 못하는 부분 제거



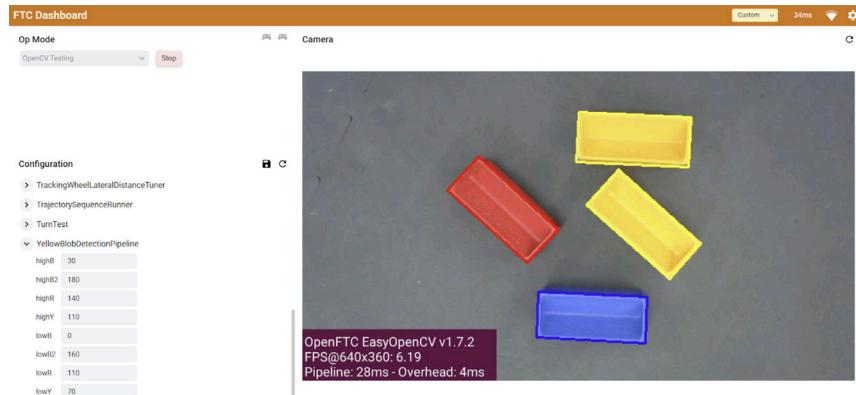
7. 윤곽에 대하여 최소 면적의 회전 직사각형 계산



8. 얻어진 직사각형으로부터 중심과 각도를 도출

첫 테스트는 코드 작성이 편리한 Python에서 진행했으며, Python에서 잘 작동함을 확인한 후 Java 코드로 옮겨 테스트를 진행함. Java의 'imageproc' 모듈과 'easyopencv' 모듈을 활용하여 웹캠에서 받아오는 실시간 영상을 처리하여 샘플을 인식하도록 코드를 작성함.

이후 Control Hub에 웹캠을 연결하여 테스트를 진행하였음. HSV 범위는 상황에 따라 달라질 수 있기에 FTC Dashboard를 활용하여 소스코드를 새로 빌드하지 않더라도 값을 수정하고 테스트할 수 있도록 구성함. 테스트 결과와 Python에서 실행했던 환경과 마찬가지로 샘플만을 정확하게 인식할 수 있음을 확인하였음.



다만, 색상의 차이를 기반으로 한 방식인 만큼, 같은 색의 샘플이 붙어있으면 두 샘플을 분류하지 못하는 문제가 발생하였음. 같은 색 끼리 묶어서 하나의 그룹으로 인식하는 방식을 사용하였기 때문에, 같은색 샘플이 2개 붙어있는 경우에 하나의 큰 사각형으로 인식하였음. 이를 해결하기 위해 2가지의 방안을 고안함.

1) 노이즈 제거 연산의 간소화

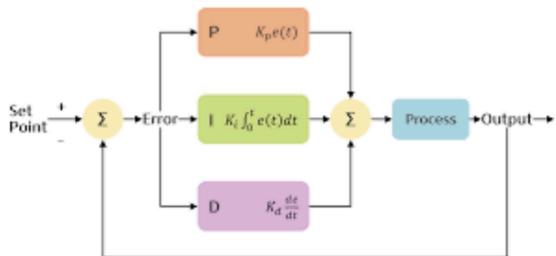
마스크의 노이즈를 제거하기 위해 튀어나온 부분을 제거하는 Open연산과 들어간 부분을 제거하는 Close연산이 활용함. 하지만 가까이 있는 2개의 샘플을 구분하려면 그 사이에 있는 민감하게 인식해야 할 필요성이 있으나, 픽셀 사이의 공간이 굉장히 얇기 때문에 Close연산을 통해 제거됨. 따라서 이 연산을 하지 않음으로써 가까이 있는 2개의 샘플을 구분할 수 있음.

2) 크기를 기반으로 한 구분

두 샘플이 완전히 붙어있어 그 사이의 틈새를 인식하기 어려운 경우에는 위의 방법으로 샘플을 구분할 수 없음. 하지만 카메라가 지면으로부터 항상 같은 거리 떨어져 지면을 수직으로 비춘다는 가정 하에, 샘플 하나가 만드는 사각형의 크기는 동일하므로, 사각형의 크기를 비교하여 특정 부분에 개의 샘플이 겹쳐있는지 확인할 수 있음.

5. PID 제어기

PID 제어기(Proportional-Integral-Derivative Controller)는 제어 시스템에서 널리 사용되는 피드백 제어 방식으로, 목표값(설정값)과 실제값(측정값) 간의 오차를 최소화하기 위해 사용됨. PID 제어기는 세 가지 구성 요소(비례, 적분, 미분)를 결합하여 시스템의 동작을 제어함. 로봇에서 정밀하게 움직여야하는 부분은 주로 서보 모터를 사용하지만, 서보 모터는 힘과 속도에 한계가 있어서 리니어를 움직이거나 무거운 팔을 회전시키는 동작에는 필연적으로 DC 모터를 사용해야함. DC 모터는 서보 모터와 다르게 전압에 기반해 속도가 조절되며, 원하는 위치로 회전시키기 위해서는 직접 엔코더 값을 활용하여 모터의 전압을 조절해야함. 또한, DC 모터는 브레이크가 없기에 외력을 무시하고 특정 상태를 유지하기 위해서는 엔코더 값을 지속적으로 모니터링 하며 전압을 조절해야함. 이러한 과정을 빠르고 효과적으로 수행하기 위하여 PID 제어기의 사용이 필수적임.



PID는 비례(Proportional), 적분(Integral), 미분(Derivative) 3가지 매개변수를 조절하여 목표 위치를 향하기 위해 필요한 힘을 결정하는 제어 기법으로, 이에 대응하는 3 가지 변수는 k_P , k_I , k_D 이며, 시간에 대한 오류함수 $E(t)$ 에 대하여 아래와 같은 관계식을 통해 PID 값을 결정할 수 있음.

$$MV(t) = k_P E(t) + k_I \int_0^t E(t) dt + k_D \frac{dE}{dt}$$

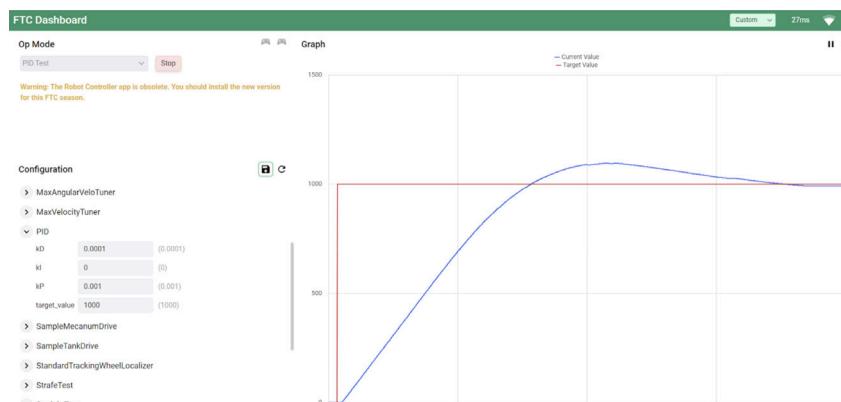


상수값들의 의미

kP | 현재 상태에서의 오차값의 크기에 비례한 제어작용을 수행

kI | 정상상태(Steady-State) 오차를 없애는 작용을 수행함

kD | 출력값의 급격한 변화에 제동을 걸어 오버슈트을 줄이고 안정성을 향상



FTC Dashboard에서 현재 엔코더 값 (current_position)과 목표 엔코더 값 (target_position)의 변화를 그 랙프를 통해 시각적으로 확인하면서 k값을 조절해보았음. 그 결과 k값을 적절하게 조절하여 빠르고 정확하게 목표 값에 도달하는 PID 제어기를 설계할 수 있었음.

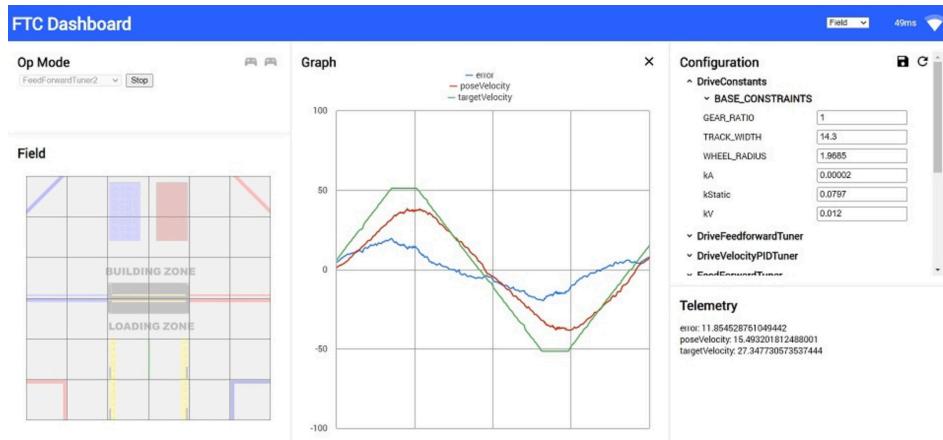
6. 시각화

프로그래밍 시에 로봇의 상태를 시각적으로 확인하여 효과적으로 소스코드를 작성하고 수정할 수 있도록 FTC Dashboard 사용법을 공부함. FTC Dashboard는 ACME Robotics에서 제작한 로봇 모니터링 웹 앱으로, 아래와 같은 다양한 기능을 제공하는 강력한 툴임.



FTC Dashboard의 기능

-] Telemetry를 통한 그래프 플롯과 필드 그라피 제공
-] 실시간 구성 변수 설정 및 확인
-] 카메라 스트리밍
-] 원격 OpMode 제어 및 게임패드 지원



FTC Dashboard를 Road Runner와 PID 제어기, 그리고 Vision을 테스트 할 때 사용하며 사용법을 익히고 유용하게 활용할 수 있도록 공부함. 다양한 기능으로 테스트 과정에서도 큰 도움이 되었으며, 로봇 소스코드 작성 과정에도 유용하게 쓰일 수 있으리라 기대됨.

FTC Dashboard 외에도 Road Runner에서 로봇의 경로 설정시에 유용하게 활용할 수 있는 MeepMeep을 설치하고, 사용법을 익힘. MeepMeep은 Noah Bres가 만든 Road Runner Trajectory 시각화 프로그램으로, trajectory 코드를 작성하여 집어넣으면 그 경로를 시뮬레이션 하여 보여줌. 로봇의 자율 주행 코드를 작성할 때, 직접적인 로봇의 주행 없이 경로를 빠르고 안전하게 확인할 수 있는 프로그램이라 생각함.

II. 코드 구조 기획

코드를 효율적으로 작성하고, 유지 보수 및 가독성을 높이기 위하여 코드 구조를 기획함. 특히 역할에 따라 파트(Part), 기능(Feature), 그리고 OpMode로 세분화하여 객체 구조를 디자인함.

1. 파트 (Part)

전체 로봇은 그 작동 방식이 복잡하기에 하나의 객체가 관리하기에는 무리가 있음. 따라서 우리는 로봇을 그 역할에 따라 나누었으며, 이 각각을 관리하는 객체를 'Part Class'라 명명하였음.



Part가 되기 위한 조건!

1. **파트의 동작은 다른 파트들과 독립되어야 함.**
2. 한 파트는 3개에서 6개 정도의 하드웨어를 제어하는 **적당한 규모**여야 함.
3. 해당 파트의 **하드웨어 제어 권한**을 전적으로 그 파트 객체에 할당할 수 있어야 함.

파트는 할당된 하드웨어의 동작을 전적으로 관리하며, OpMode에 의해 통제될 수 있는 적절한 "명령어 함수"를 제공함. 이를 통해 객체 지향 프로그래밍(OOP)의 다형성(Polymorphism), 캡슐화(Encapsulation), 그리고 추상화(Abstraction)을 달성하여 코드의 관리를 용이하게 함.

2. 기능 (Feature)

다양한 파트 및 OpMode에서 사용할 수 있는 유용한 기능들을 구조화하여 객체로 제작함. 이 객체들은 그 기능에 따라 다양한 형식과 형태를 가질 수 있음. 여러 곳에서 사용될 수 있는 기능들을 객체화하여 객체 지향 프로그래밍 (OOP)의 주된 특성인 캡슐화(Encapsulation)를 달성하여 코드의 중복 사용과 그로 인한 실수를 방지함.

3. OpMode

로봇 컨트롤러 코드의 가장 중추적인 부분으로, 자율 주행 시기의 AutoOp와 원격 조정 시기의 TeleOp 코드를 작성하는 부분임. OpMode에 해당하는 객체들은 각 파트들을 통합하여 전체적인 로봇의 움직임을 총괄 및 제어하지만, 이들은 각 파트가 지원하는 "명령어 함수"를 실행할 수 있는 권한만 가질 뿐, 하드웨어를 직접적으로 제어할 수 없음.

기능을 분리하여 객체를 생성하고, 이들 사이의 유기적인 관계를 구성한 객체지향적인 프로그래밍을 통하여 관리하기 쉽고, 오류를 구조적으로 예방할 수 있는 코드 구조를 구축함.



기존 방식과의 차이점!

1. 하드웨어 객체

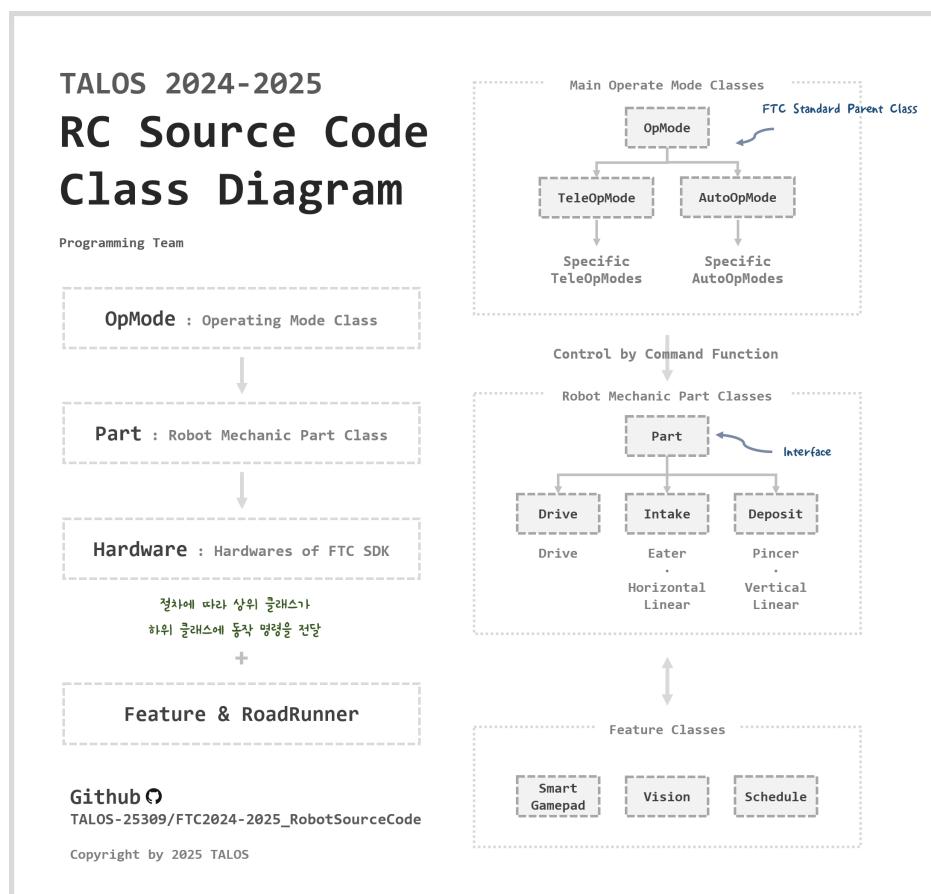
기존의 방식은 하드웨어 객체를 새로 정의하여 파트에서의 하드웨어 사용을 용이하게 하였으나, 파트가 하드웨어를 직접적으로 조작할 수 없어, 그 기능이 제한되고 하드웨어 객체가 복잡해지는 문제점이 있었음. 이로 인하여 이번 시즌 코드에서 하드웨어 객체 레벨은 제거됨.

2. 파트 객체의 명령어 작동 방식

현재 파트 객체가 지원하는 “명령어 함수”는 Java의 기본 메소드 형태이며, 즉각적인 명령을 지원하고, 필요한 경우에만 Update 함수를 통한 후처리를 지원함. 하지만 기존 파트 객체는 비동기 프로그래밍에 중점을 두어 문자열 형태의 명령어를 전달하고, 명령을 예약했다가 Update 함수에서 이를 처리하는 방식을 활용하여 오버헤드 발생, 잘못된 명령어 실행, 실행시간 지연 등의 문제가 발생하였음. 이번 시즌에는 이러한 문제를 해결하여 소스코드의 구조를 새로 디자인함.

3. 기능 객체

기능 객체 레벨을 새로 개발함. 이로 인하여 기존에 비해 중복되는 기능에 대한 코드 재사용이 줄어들었으며, 객체를 통한 체계적인 관리로 코드의 가독성을 높이고, 효율을 극대화 시킴.



III. 파트 객체 개발

1. 파트 인터페이스 및 기본 구조 개요

모든 파트는 파트 인터페이스 (Part Interface)를 상속받아야 함. 파트 인터페이스는 파트에 필수적으로 존재해야하는 함수를 강제하며, 이를 통해 다형성 (Polymorphism)을 구현하여 OpMode에서 공통된 기능을 쉽게 접근할 수 있도록 제작함. 이에 해당하는 함수는 아래와 같음.



공통된 함수 목록!

Init | 초기화 함수, Hardware Map과 Telemetry를 매개변수로 받음

Update | 주기적으로 확인해야하는 작업을 진행하는 업데이트 함수

Stop | 비상 상황에서 모든 기능을 정지하고 수동 모드에 진입시킬 수 있는 함수

그 외에 모든 파트는 각자 고유한 일반 함수와 “명령어 함수”를 가질 수 있음. 고유 함수는 내부에서 사용되는 Private 함수이며, “명령어 함수”는 외부에서 명령을 내릴 수 있도록 파트가 제공하는 함수로, Public 함수이며 함수명이 cmd[명령어]로 구성됨.

모든 파트는 3가지 종류의 클래스에 의해 정의됨.

메인 파트 클래스

Main Parts Class

서브 파트 클래스

Sub Parts Class

파트 상수 클래스

Part Constants Class

모든 파트는 파트 명에 해당하는 메인 파트 클래스를 Public 클래스로 가지며, 외부에서는 이 메인 파트 클래스에만 접근할 수 있음. 그 외에는 파트를 작게 쪼개어 객체화한 Default 클래스인 서브 파트 클래스가 존재하며, 서브 파트 클래스들은 메인 파트 클래스에 의해 제어됨. 대부분의 경우 하드웨어는 해당 서브 파트 클래스가 제어하며, 서브 파트 클래스가 메인 파트 클래스를 위한 “명령어 함수”를 제공하고, 메인 파트 클래스는 서브 파트 클래스의 “명령어 함수”를 통합한 고유의 “명령어 함수”를 가짐. 파트 내의 유기적인 연결과 로봇이 구동중인 상태에서 빠르게 상수를 바꿀 수 있도록 파트 상수 클래스를 제작하였으며, 내부의 값은 정적(Static) 값으로 모든 클래스가 그 값을 공유함.

2. Drive Part

로봇의 구동부를 관리하는 파트 클래스임. Gamepad를 통한 로봇의 병진 및 회전 운동을 관리하며, Road Runner를 사용한 Localization과 Following Trajectory를 지원함.

메카님 휠을 세밀하게 조정하여 상하좌우 뿐만 아닌 모든 방향으로의 자유로운 움직임을 지원하며, 회전과 병진 운동을 동시에 수행할 수 있음. 뿐만 아니라 Localization을 활용하여 로봇의 방향과 관계 없이 절대적인 좌표계에서 움직일 수 있는 필드 중심 주행을 지원한다는 점에서 차별화된 강력한 주행 시스템을 제공함.

1) Class 구조

클래스	구분	역할
Drive	Main Part	구동부 및 Road Runner 관리
DriveConstants	Part Constants	속도, 위치 데이터 등 구동 속성값 관리

2) 명령어 목록

명령어 함수	매개 변수	기능
Drive	x, y, omega	입력을 기반으로 주행
DriveSlowly	x, y	병진 운동을 통한 미세 조정 지원
AutoAlignBasket		그물(Basket) 위치로 이동하는 매크로
AutoAlignSpecimen		관찰 구역으로 이동하는 매크로
AutoAlignSubmersible		Vision을 활용한 Intake 자동 조준 매크로

3. Intake Part

Sample과 Specimen을 습득하는 장치 (Intake)를 관리하는 클래스임. 엔코더를 활용한 수평 리니어 제어와 서보 모터 제어를 지원함. 수평 리니어는 PID 제어기기를 활용하여 특정 위치에 고정될 수 있도록 하였으며, 이터는 서보의 회전 범위를 파트 상수 클래스를 통하여 지정하여 세밀하게 움직임을 제어함.

조작을 간편히 하기 위해 습득 과정을 4가지 단계로 구분함. currentStep 변수에 현재 단계를 저장하고, 단계에 따라 (0) ~ (3) 의 함수를 실행함. 실행 순서는 선형적이지 않기 때문에 runNextStep와 runPrevStep 함수로 단계를 변경함.

1) Class 구조

클래스	구분	역할
Intake	Main Part	Intake 전체 관리 및 총괄
IntakeConstants	Part Constants	리니어 엔코더 값, 서보 범위 등 관련 속성값 관리
Eater	Sub Part	샘플을 빨아들이는 이터의 작동 및 움직임 관리
HorizontalLinear	Sub Part	PID를 사용한 수평 리니어의 움직임 제어

2) 명령어 목록

명령어 함수	매개 변수	기능
(0) StretchLinear		리니어 확장 및 리니어의 자유로운 조작
(1) Eat		이터 하강 및 샘플 습득
(2) Vomit		샘플 제거 및 이터 상승
(3) Transfer		Intake에서 Deposit으로 전달
runPrevStep		이전 step의 함수 실행
runNextStep		다음 step의 함수 실행
AutoStretch		정해진 위치로 수평 리니어 확장
AutoRetract		로봇 안으로 들어오도록 수평 리니어 수축
ManualStretch		컨트롤러의 명령에 따른 수동 리니어 확장
ManualRetract		컨트롤러의 명령에 따른 수동 리니어 수축
ManualStop		수동 상태에서 PID로 리니어 고정
EaterRun		이터 회전을 통한 샘플 습득
EaterStop		이터 정지
AutoRotate		샘플 감지에 따른 이터 회전
ManualRotate	direction	컨트롤러의 명령에 따른 이터 회전
ArmUp		이터 상승
ArmDown		이터 하강
runCurrentStep		현재 step에 맞는 함수 실행

4. Deposit Part

Intake에서 전달받은 Sample과 Specimen을 각각 Basket과 Chamber에 놓고 오는 클래스. 엔코더를 활용한 수직 리니어 제어와 서보 모터 제어를 지원함. 수직 리니어는 PID 제어기기를 활용하여 특정 위치로 정확히 이동할 수 있도록 하였으며, 집게는 고정된 회전 범위에서 운동함.

1) Class 구조

클래스	구분	역할
Deposit	Main Part	Intake 전체 관리 및 총괄
DepositConstants	Part Constants	리니어 도달 위치 등 관련 속성값 관리
Claw	Sub Part	샘플을 운반하는 집게의 움직임 관리
VerticalLinear	Sub Part	PID를 사용한 수직 리니어의 움직임 제어

2) 명령어 목록

명령어 함수	매개 변수	기능
DepositSample	location	샘플을 2가지 바구니 중 하나에 드롭
DepositSpecimen	location	견본을 2가지 체임버 중 하나에 드롭
Return		수직 리니어 수축 및 집게 조정
ManualStretch		컨트롤러의 명령을 통한 수직 리니어 확장
ManualRetract		컨트롤러의 명령을 통한 수직 리니어 수축
ManualStop		수동 상태에서 PID로 리니어 고정

IV. 기능 객체 개발

1. Smart Gamepad

FTC SDK에서 기본적으로 제공하는 Gamepad 객체는 각 버튼이 눌렸으면 참, 눌리지 않았으면 거짓의 값을 나타냄. 따라서 어떤 버튼이 클릭되었을 때 어떠한 명령을 처리하기 위해서는, 현재 버튼 상태가 참임을 확인하는 것 뿐만 아니라, 이전의 버튼 상태가 거짓임을 추가로 확인해야함. 이를 위하여 게임패드를 복사하고, 조건을 구성하는 것은 번거롭고 실수를 유발하기 쉬우므로, 이 기능을 객체화하여 처리함.

Smart Gamepad는 자동으로 이전 게임 패드의 상태를 저장하며, 직관적인 함수 호출을 통해 다양한 상태에 대한 참/거짓 값을 알아낼 수 있음. 또한, Getter를 활용해 등록한 게임패드에 직접 접근하여 값을 확인하는 것 역시 가능하도록 디자인 함.



Smart Gamepad의 함수!



isPressed | 버튼이 새로 눌렸는지 확인 (Now True + Prev False)

isReleased | 버튼이 떼어졌는지 확인 (Now False + Prev True)

isHeld | 버튼이 눌려있는지 확인 (Now True + Prev True)

isFree | 버튼이 눌려있지 않은지 확인 (Now False)

2. Vision

OpenCV 모델을 이용한 샘플 구분 기능을 구현한 클래스. Intake 파트에서 타겟 샘플을 찾는 과정을 단순하게 사용할 수 있도록 설계함. setTargetColor은 목표 색을 설정하고 detectTarget은 가장 가까운 샘플의 정보를 전달함. Sample의 정보를 쉽게 관리하기 위해 Sample 클래스를 만들어 위치와 각도, 색을 저장함. 웹캠 관리 및 타겟 샘플 감지는 독립적인 pipeline을 만들어서 영상 처리에 필요한 기능을 순차적으로 진행함. 상세한 과정은 Part 1 - 4.openCV 에 설명되어 있음. 카메라의 위치에 가장 가까운 샘플을 찾아내는 기능은 추가로 구현함.

전체적인 로봇 작동의 효율을 높이기 위하여 telemetry를 이용해 영상을 전달하는 기능을 최소화함. 또한 샘플을 감지할 필요가 없는 경우에 Vision 파트의 작동을 중지할 수 있도록 turnOn, turnOff 함수를 설정함.

3. Schedule

일련의 명령을 특정 시간을 간격으로 하여 처리하기 위하여 단순히 명령어를 나열하고, 그 사이에 딜레이를 넣으면, 그 시간동안 로봇이 작동 불능 상태가 되어 위험할 수 있음. 따라서 특정 일이 진행중일 때 다른 일을 동시에 처리할 수 있는 비동기적 프로그래밍을 수행해야함.

비동기 프로그래밍을 하는 가장 대표적인 방법은 Java의 쓰레드를 사용하는 것임. 그러나 쓰레드는 잘못 사용하면 오히려 성능이 떨어질 뿐더러, 메모리 동시 접근 시에 버그 및 오류가 발생할 수 있음. 또한, 하드웨어 접근은 특히 오버헤드가 많이 발생하기에, 동시에 하드웨어에 접근하면 치명적인 오류가 발생할 수 있음. 쓰레드 안전 (Thread Safety)한 시스템을 작성하는 것은 매우 어려우며, 때로는 비효율적이므로, 메인 쓰레드에서 주기적인 호출을 통해 비동기 프로그래밍을 구현할 수 있는 새로운 시스템을 객체 형태로 개발함.

Schedule은 정적 Public 클래스로, 모든 영역에서 접근 가능함. Schedule 객체는 일(Task)를 Runnable 타입의 수행할 일과 수행할 시기를 입력받아, 우선 순위 큐 (Priority Queue)에 저장하여 주기적인 Update 함수가 호출될 때, 내부 타이머를 확인하여 정확한 시기에 수행할 일을 실행시킬 수 있음. 이를 통해 시간 기반 메인 쓰레드에서의 비동기 프로그래밍을 구현함.

V. OpMode 개발

1. OpMode 객체 개요

OpMode 객체는 로봇을 작동시키는 메인 클래스로, 로봇의 전체적인 구동을 총괄함. OpMode는 Part 객체를 정의하고, 드라이버의 입력 혹은 자율주행 절차에 따라 Part의 명령어 함수를 활용하여 적절한 명령을 내림. 또한, 각각의 파트들과 Schedule 객체를 지속적으로 업데이트하면서 메인 루프 안에서 비동기 명령을 처리하는 역할을 수행함. 또한 Part에 직접적인 명령을 내리는 것 외에도 Feature 객체에 해당하는 Vision과 Smart Gamepad 객체의 정보를 통합하여 복합적인 명령을 구성하는 기능을 수행함.

2. TeleOpMode (Driver Controlled Period)

TeleOpMode 객체는 원격 조정 기간 (Driver Controlled Period) 동안 게임 패드를 통한 드라이버의 명령을 전달받아 이를 재조합하여 Part에 명령을 하달하는 역할을 수행함. 전체적인 구조는 FTC SDK의 OpMode를 상속받아 구성됨.

TeleOpMode의 가장 중심적인 기능은 컨트롤러 (게임 패드)의 입력값을 통해 드라이버의 명령을 정확히 전달받고, 이를 활용해 로봇의 전체적인 구동을 총괄하는 것임. 따라서 이 객체를 정의할 때에는 게임 패드의 각 버튼에 무슨 역할을 할당하고, 이것을 어떤 파트의 어떤 명령어 함수, 혹은 어떤 Feature의 기능을 사용할지를 정의하는 것이 중요했음.

프로그래밍 팀은 드라이버들과 긴밀히 소통하며 최대한 로봇을 조작하기 편리하고 효과적인 방법을 찾기 위해 많은 토의를 거쳤으며, 다양한 테스트를 통하여 최적의 방법을 찾고자 노력함. 이를 통해 최종적으로 컨트롤러의 각 입력에 역할을 부여하고, 이를 명령어 함수들과 연결시킴으로써 TeleOpMode 객체를 제작함.

3. AutoOpMode (Autonomous Period)

AutoOpMode 객체는 30초 동안의 자율 주행 시간 (Autonomous Period) 동안 로봇이 자율적으로 구동할 수 있도록 하는 일련의 명령을 관리하는 객체임. 기본적으로 주어진 절차에 따라 명령을 순차적으로 실행하며, 각종 센서와 카메라 입력값을 활용하여 정확하게 명령을 실행하는 것을 목표로 함. 전체적인 구조는 FTC SDK의 LinearOpMode를 상속받아 구성됨.

AutoOpMode의 가장 기본적인 구조는 Road Runner의 Trajectory Sequence 생성의 형태를 가짐. Trajectory Sequence를 빌드하고 실행시킴으로써 정확한 위치로의 자율 주행을 구현하고, 그 사이에 마커를 삽입하여 Drive 파트를 제외한 다른 파트에 명령을 전달함으로써 자율적인 로봇 구동을 달성함. 만들어진 Trajectory Sequence는 Drive 파트에 의해 실행됨.



Controller Guidebook DRIVER #1 : Drive



TIP

- 기본적인 이동은 조이스틱 활용
- 점수정에서 샘플을 조준할 때에는 미세 조종 추천
- 매크로는 목표 지점 가까이 이동한 후 사용
- 리니어 당기기를 누른 후에는 행잉 모드에 돌입하므로 주의




Controller Guidebook DRIVER #2 : Intake & Deposit



TIP

- Intake 스텝 조절 특히 주의하여 작동
- Deposit 버튼 헛갈리지 않게 정확하게 속지
- 특이한 경우가 아니라면 오토 모드를 사용, 수동 모드는 자양




Controller Guidebook EMERGENCY MODE



Emergency Mode 실행
(4개의 버튼을 함께 클릭, 둘 중 한 명만 눌러도 실행)



Emergency Mode

위험 상황 발생 시, 고급 정지용 기능
PID와 Road Runner에서 문제가 발생할 수 있으니, 이 경우 긴급하게 작동시켜야 함.



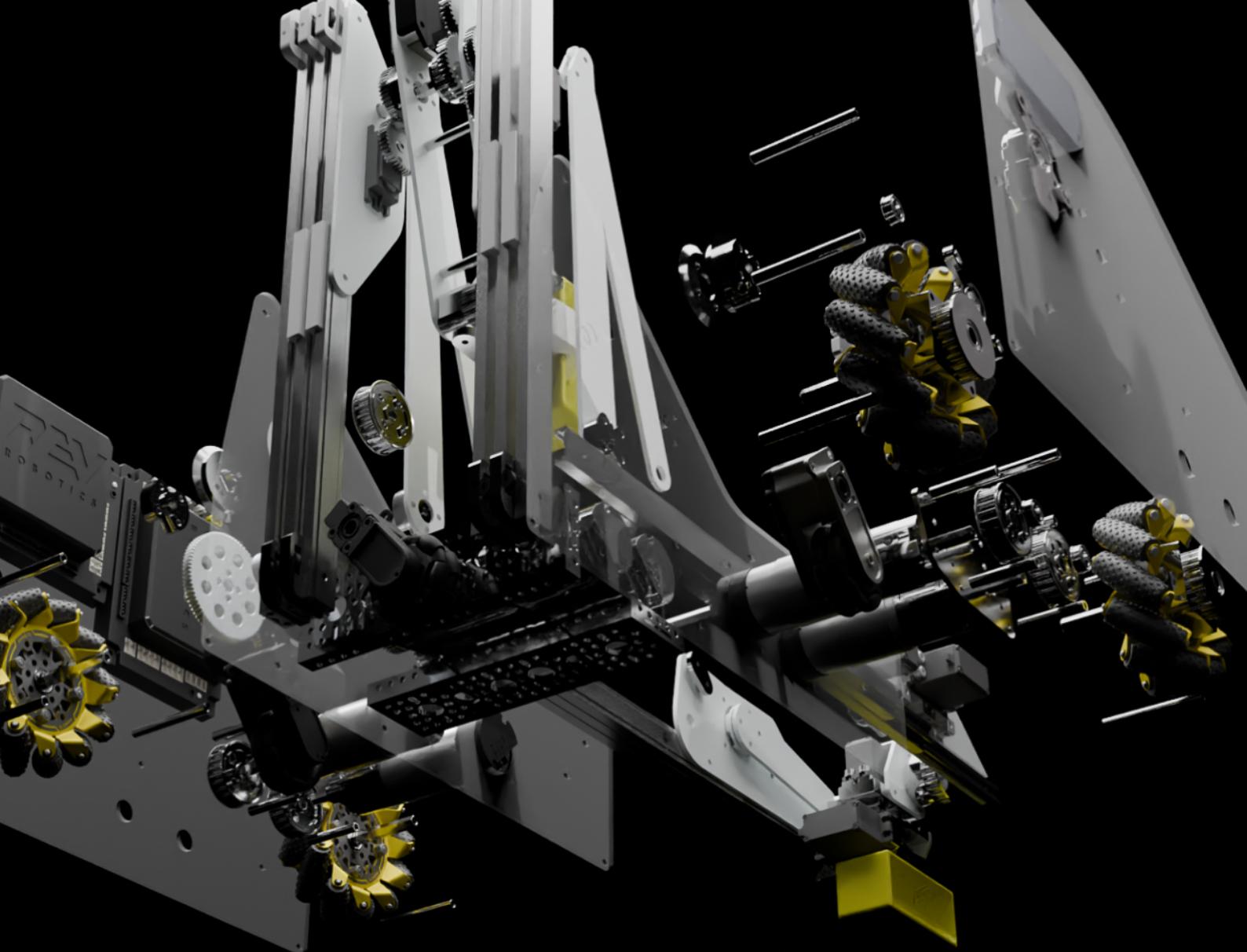
1) [전략 1] Basket에 Sample 넣기 전략

이 전략은 로봇이 중립 샘플 가까이에 배치되었을 때 사용하는 전략으로, 로봇에 사전에 로딩되어있는 샘플을 먼저 가장 높은 Basket에 넣고, 그 뒤로 바닥에 있는 3개의 중립 샘플을 순차적으로 집어 Basket에 넣는 것을 목표로 함. 또한, 이 모든 과정이 끝나면 잠수정에 붙어 주차를 수행함.

2) [전략 2] Chamber에 Specimen 걸기 전략

이 전략은 로봇이 자신의 컬러에 해당하는 샘플 가까이에 배치되었을 때 사용하는 전략으로, 로봇이 사전에 로딩되어있는 견본을 먼저 높은 Chamber에 건 후, 바닥에 배치된 모든 컬러 샘플을 순차적으로 관찰 구역으로 밀어넣어 견본으로 만들어 최종적으로 모든 견본을 높은 Chamber에 거는 것을 목표로 함. 또한, 이 모든 과정이 끝나면 잠수정에 붙거나 혹은 상황에 따라 관찰자 구역으로 들어가 주차를 수행함.

이 과정에서는 Road Runner의 정확한 구동이 필수적임. 따라서 이를 위해서는 정확한 튜닝과 함께 정확한 경로 설정이 뒷받침 되어야 함. 이를 위하여 프로그래밍 팀은 많은 시간을 투자하여 어떠한 전략이든 최대 점수를 달성하기 위해 최선을 다함.



#25309
TALOS

탈로스는 그리스 신화 속 대장장이
헤파이스토스가 만들어 낸 최초의 로봇입니다.

TALOS라는 팀 명칭은 헤파이스토스가 이성을
통해 창조한 탈로스처럼 멋진 로봇을 만들어
내겠다는 의지를 표방합니다.