

로봇팔 교재

로봇 스튜디오 창 하나에서 AI 입문, 탁구공을 스스로 잡는 로봇팔, 로봇에게 가르치는 모방학습, 레일 로봇, 드로잉, 로봇 농구 경기까지 차례대로 따라 합니다. [다른 교재 보기](#)

목차

1. **이 교재로 하는 일** · 하고 싶은 것으로 단원 찾기
2. **로봇 알아보기** · 장치 · 서보와 스텝
3. **시작하기 — 로봇 스튜디오** · 15분
4. **안전 수칙**
5. **AI 하드웨어 입문 (Start_AI)** · 2~6차시
6. **6축 로봇팔 — 미니 로봇팔로 조종** · 1차시
7. **나만의 그리퍼 만들기** · 2차시
8. **탁구공 로봇팔 — 키보드 · 손 · AI 자동** · 2차시
9. **모방학습 — 로봇에게 가르치기** · 2~3차시
10. **레일 로봇팔** · 1차시
11. **그림 그리는 로봇** · 2차시
12. **로봇 농구 경기** · 1차시
13. **컨베이어 벨트** · 1차시
14. **디지털 트윈** · 30분
15. **안 될 때**
16. **부록 A. AI에게 물어보기**
17. **부록 B. 더 보고 싶을 때**

이 교재로 하는 일

로봇 교구는 장치가 여러 개입니다. 모두 **로봇 스튜디오** 창 하나에서 열 수 있습니다. 하고 싶은 것을 찾아 그 단원부터 해도 되고, 처음이면 목차 순서대로 하면 됩니다.

로봇팔 및 경기장 · 10가지 프로젝트

만들고 · 제어하고 · 가르치고 · 성장한다



1



6축 로봇팔

마스터 레버(포텐쇼미터)로
슬레이브 서보 6개를 조종

Arduino Nano · 마스터-슬레이브

2



AI 하드웨어 입문(Start_AI)

Arduino+웹캠+서보로
AI 제어 첫걸음 (Firmata)

Start_AI · servo_test.py

3



로봇 경기장 점수판

거리 센서로 홀인 감지,
도트 매트릭스 자동 점수판

ESP32 · VL53L0X · MAX7219

4



탁구공 잡기 (스테퍼)

YOLO로 공 감지 → 스텝퍼
로봇팔이 잡아 클레에

CNC Shield · GRBL · YOLOv8

5



탁구공 잡기 (서보)

같은 자동 잡기를
서보 로봇팔로 구현

PyFirmata · 서보 · YOLOv8

6



3모드 로봇팔-레일

PS4.키보드.손으로
로봇팔/레일 직접 제어

GRBL G-code · 서보/스텝퍼

7



그림 그리는 로봇

얼굴 SVG를 G-code로 변환
로봇팔이 외곽선 드로잉

MediaPipe · face_to_gcode.py

8



컨베이어 벨트

서보 투하 → 벨트 이동 →
로봇팔 전달 자동 시퀀스

Arduino Uno · TMC2208 · LCD

9



레일 로봇 교육

4축 레일 로봇으로 직접 제어
+ 모방 학습 + YOLO 자동 (6모드)

education_rail_robot

10



로봇팔 모방 학습

사람 시연을 보여주면 학습
손짓.키보드.회귀.KNN 4종

imitation_learning_robot

로봇팔 및 경기장의 10가지 프로젝트 (photo_robot_projects.png)

하고 싶은 것	단원	폴더
프로그램으로 LED · 서보를 처음 움직여 보고 싶다	AI 하드웨어 입문	2. Start_AI
레버를 돌리면 따라 움직이는 로봇팔	6축 로봇팔	1. arduino
공을 잘 잡는 나만의 로봇 손	나만의 그리퍼	3D 설계 · 출력
AI가 스스로 탁구공을 잡게 하고 싶다	탁구공 로봇팔	4. (스텝) · 5. (서보)
로봇에게 잡는 법을 "가르치고" 싶다	모방 학습	10. (서보 · 스텝)
레일로 더 넓은 범위를 다루고 싶다	레일 로봇팔	6_Robot_Stepping_Rail · 9.
로봇팔이 내 그림을 그리게 하고 싶다	그림 그리는 로봇	7. drawing
로봇 농구 경기를 하고 싶다	로봇 농구 경기	3. baseballstage_esp32
공장처럼 자동 공급 라인을 만들고 싶다	컨베이어 벨트	8. conveyor
화면 속 3D 로봇팔이 실물을 따라 움직이게	디지털 트윈	11. digital_twin

처음이라 뭘 골라야 할지 모르겠으면

AI 하드웨어 입문의 1단계(LED)와 4단계(서보)만 먼저 하세요. "내 코드가 실물을 움직인다"는 경험이 30분 안에 생기고, 그다음부터는 어느 단원으로 가든 무섭지 않습니다.

교재 속 화면 그림에 대해

화면 그림은 스튜디오를 **연습 모드(가짜 로봇)**로 실제 실행해 찍었습니다. 카메라 자리에는 **그려 넣은 탁구공 작업판과 AI로 만든 손 그림**을 넣었습니다. 그래서 창 위에 "연습 모드"라고 적혀 있습니다. 내 화면은 실제 카메라 영상이 나옵니다.

로봇 알아보기



로봇팔 및 경기장 (photo_robot_overview.png)

로봇팔 제작 키포인트

서보 모터 & 가이드
서보 모터 (마리)를 오른쪽으로 배치하고, 전선을 가이드 홈에 넣어 정리한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

그리퍼 결합부
나사나 스톱으로 손잡이 장착 위치를 잡고, 스톱으로 손잡이를 고정한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

■ 완성 모델 (3D 설계)
이제 3D 모델로 설계된 로봇팔을, 서보 모터와 3D 프린팅 출력물을, 베이킹으로 제작한다.
많은 학생이 지속적으로 사용하고, 무거운 물건을 옮길 때에도 테스트 하기 때문에 비를 줄이기 위하여 내구성이 뛰어난 서보모터를 사용하는 것을 추천한다. 키포인트 2번 서보모터는 가장 힘이 집중되는 곳으로 3kg 이상의 부하를 내는 서보모터를 권장한다.

로봇팔 (photo_arm.png)

미니 로봇팔 제작 키포인트

서보 모터 & 가이드
서보 모터 (마리)를 오른쪽으로 배치하고, 전선을 가이드 홈에 넣어 정리한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

그리퍼 결합부
나사나 스톱으로 손잡이 장착 위치를 잡고, 스톱으로 손잡이를 고정한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

서보 모터 방향 (오른쪽)
해당 관절 서보 모터의 마리를 오른쪽 방향으로 배치한다.

서보 모터 방향 (왼쪽)
반대쪽 관절 서보 모터의 마리를 왼쪽 방향으로 배치한다.

■ 완성 모델 (3D 설계)
이제 3D 모델로 설계된 로봇팔을, 서보 모터와 3D 프린팅 출력물을, 베이킹으로 제작한다.
많은 학생이 지속적으로 사용하고, 무거운 물건을 옮길 때에도 테스트 하기 때문에 비를 줄이기 위하여 내구성이 뛰어난 서보모터를 사용하는 것을 추천한다. 키포인트 2번 서보모터는 가장 힘이 집중되는 곳으로 3kg 이상의 부하를 내는 서보모터를 권장한다.

미니 로봇팔(마스터) (photo_mini_arm.png)

서보와 스텝 — 두 갈래

탁구공 로봇팔과 모방학습 로봇에는 **서보 버전**과 **스텝(스테퍼) 버전**이 있습니다. 같은 일을 두 방식으로 해 보며 차이를 몸으로 배웁니다.

	서보 버전	스텝 버전
움직이는 방법	각도를 바로 지시 (PyFirmata)	위치를 지시 (G-code, GRBL)
성격	즉시 반응 · 코드가 단순	정밀 · 경로 계획
폴더	5. · 10. ..._servo	4. · 10. ... - stepping
레이 확장	안 됨	됨 (6_Robot_Stepping_Rail · 9.)

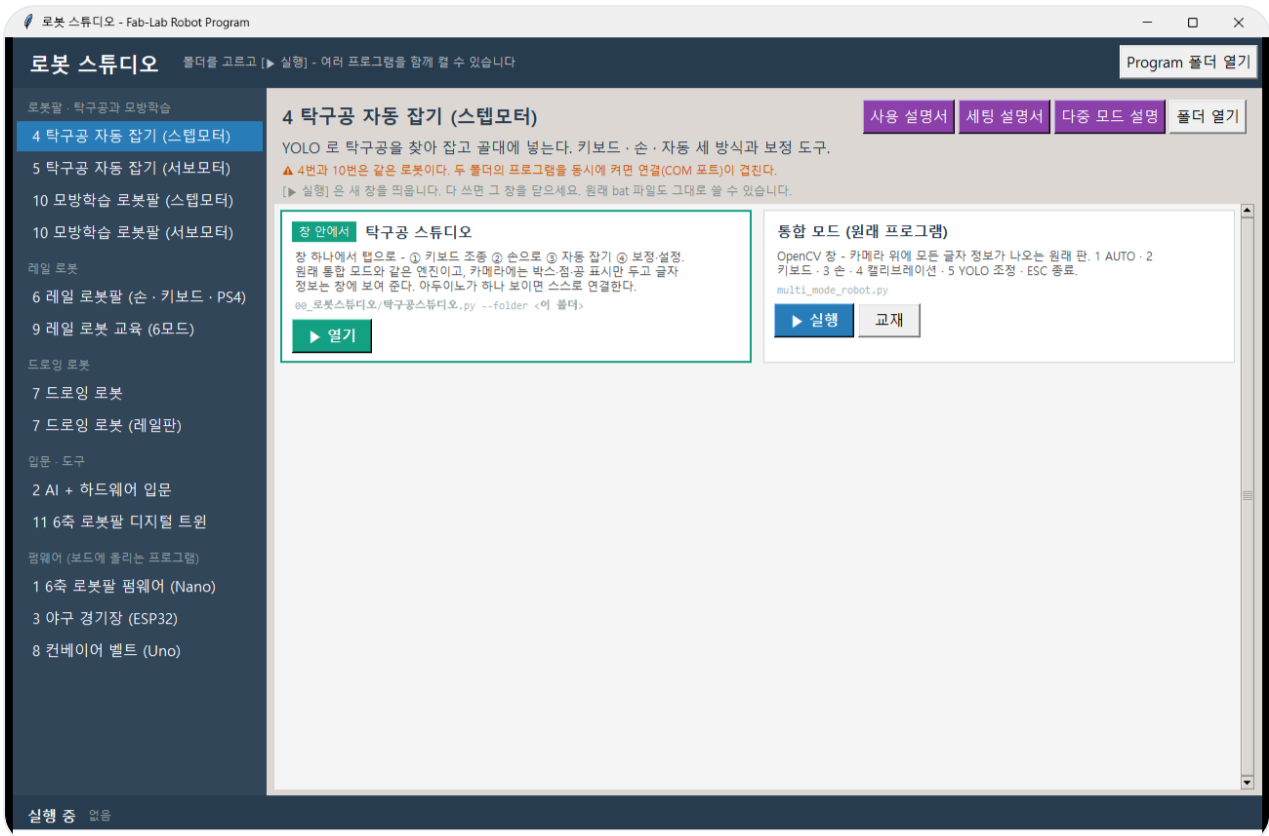
처음이면 **서보 버전**이 간단해서 좋고, 정밀한 반복 작업과 레일은 **스텝 버전**의 무대입니다. 두 버전은 화면과 키가 똑같습니다.

시작하기 — 로봇 스튜디오 열기

시간 15분 · 이번에 하는 일 로봇 스튜디오를 열고, 카드로 프로그램을 띄우는 법과 로봇 보드의 포트 번호 확인법을 익힌다.

(가) 로봇 스튜디오를 연다

파일 탐색기 → `D:\data\study\AI\Fab-Lab\Robot\Program` → `로봇스튜디오_시작.bat` 더블클릭



로봇 스튜디오. 왼쪽에서 폴더를 고르면 오른쪽에 그 폴더의 프로그램 카드가 나옵니다 (screen_hub.png)

- 1 왼쪽 목록에서 하고 싶은 폴더를 누릅니다. 예를 들어 탁구공 로봇팔은 **4 탁구공 자동 잡기 (스텝모터)** 입니다.
- 2 오른쪽 카드의 **▶ 열기** 또는 **▶ 실행** 을 누르면 그 프로그램이 **새 창**으로 뜹니다.
- 3 다 쓰면 **그 창을 직접 닫습니다**. 로봇이 정리하고 끝나므로 이 방법이 가장 안전합니다.

(나) 카드는 두 종류입니다

카드	무엇

초록 '창 안에서' 카드 ▶ 열기	스튜디오 모양의 창. 왼쪽 카메라, 오른쪽 한글 단추와 설명. 이 교재는 이 창으로 합니다
파란 카드 ▶ 실행	원래 프로그램(영어 글자가 카메라 위에 뜨는 창). 키는 같습니다

폴더 하나만 쓸 때는 그 폴더 안의 **스튜디오_시작.bat**(또는 **탁구공스튜디오_시작.bat** 같은 이름)을 바로 더블클릭해도 됩니다.

(다) 보드의 포트(COM) 번호 확인하기

스튜디오는 아두이노가 **하나** 꽂혀 있으면 스스로 연결합니다. 여럿이거나 안 붙으면 포트를 직접 고릅니다.

1 보드를 USB로 꽂고 Win + X → 장치 관리자를 엽니다.

2 포트(COM & LPT)를 펼칩니다. USB-SERIAL CH340(COM5) 처럼 보이면 괄호 안 **COM5**가 포트 번호입니다.

목록에 아무것도 없으면

충전 전용 케이블이 아니라 **데이터 케이블**인지, CH340 드라이버가 깔려 있는지 확인하세요. USB를 뽑았다 꽂을 때 사라졌다 나타나는 항목이 그 보드입니다.

(라) 같이 쓸 때 주의

- 같은 로봇을 두 프로그램이 함께 쓰면 둘 다 안 됩니다(COM 포트가 겹칩니다). 4번 탁구공과 10번 모방학습(스텝)은 같은 로봇입니다.
- 다른 로봇끼리는 함께 켜도 됩니다(예: 컨베이어 + 탁구공 스튜디오 + 디지털 트윈).

안전 수칙 — 로봇을 다루는 사람의 기본기

- 로봇은 전원을 넣거나 연결하는 순간 **갑자기 움직일 수 있습니다**. 연결하면 처음 자리(홈)로 움직입니다. 팔이 닿는 곳을 비우고 손·얼굴을 넣지 않습니다.
- 이상하면 창 맨 아래 **[■ 비상 정지]**를 누릅니다. 모터에 힘이 빠지므로 **팔이 처질 수 있습니다**. 다시 쓰려면 끊기 뒤 연결.
- 보드와 센서의 **전압 표기(3.3V / 5V)**를 반드시 확인합니다. 경기장 거리센서(VL53L0X)는 3.3V를 넘으면 망가집니다.

4. 서보 여러 개는 **외부 전원**을 씁니다. 아두이노 USB 전원만으로 서보 5~6개를 돌리면 떨리고 리셋됩니다.

AI 하드웨어 입문 — Start_AI 8단계

시간 4~6차시(핵심 1 → 4 → 7 → 8단계만이면 2차시) · 이번에는 아두이노 + 파이썬으로 LED부터 AI 비전까지 8단계를 오른다.

Start AI 키트 제작 키포인트

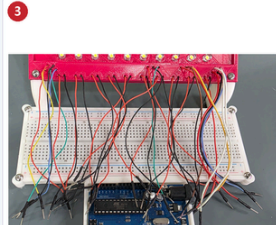
1 제작 키포인트



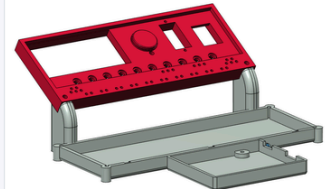
LED 10개 개별 연결
LED 10개를 개별적으로 연결하여, 인공지능 또는 PWM 신호에 반응하는 모습을 관찰할 수 있도록 한다.



릴레이 배선
릴레이 부분의 배선은 출력 단자의 선 색을 입력 단자와 별도로 구분하여 연결한다.



브레드보드 배선 길이
브레드보드를 활용할 수 있도록 선 길이를 너무 길지 않게 한다. 선이 너무 길면 서로 꼬여 정리가 어렵다.



■ 완성 모델 (3D 설계)
전자회로를 처음 접하는 학생에게 적합한 키트로, 짧은 실습 시간을 최대한 활용하도록 정리정돈의 효율을 높이고 주요 전자 부품의 작동 시험을 할 수 있도록 구성하였다.

아두이노와 파이썬을 활용한 기초 AI 활용 수업용 프로그램을 함께 제공하여, 학생이 쉽게 적용할 수 있도록 하였다.

AI 기초보드(Start AI) 구성 (photo_startai.png)

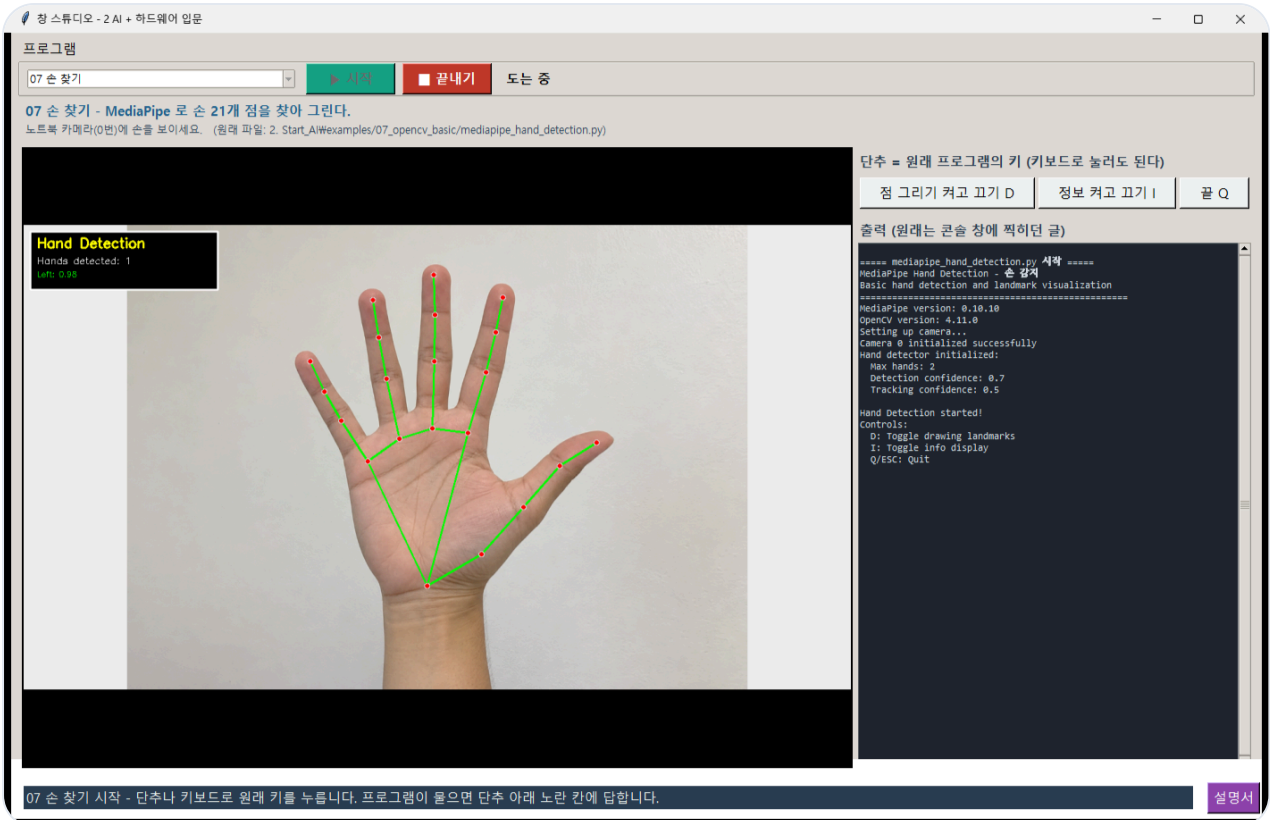
(가) 처음 한 번 — StandardFirmata 올리기

파이썬이 아두이노 핀을 직접 다루려면 아두이노에 통역사(StandardFirmata)를 한 번 올려 둡니다.

- 1 **Arduino IDE**를 엽니다(없으면 arduino.cc 에서 Windows용을 받아 설치).
- 2 아두이노 우노를 USB로 꽂고 메뉴 **파일 → 예제 → Firmata → StandardFirmata**.
- 3 **도구 → 보드 = Arduino Uno**, **도구 → 포트 = 연결된 COM**을 고르고 업로드(→ 단추). "업로드 완료"가 뜨면 끝입니다.

(나) 입문 예제 스튜디오로 실행한다

Robot\Program\2. Start_AI → **입문예제스튜디오_시작.bat** (또는 로봇 스튜디오 → 2 AI + 하드웨어 입문 → 초록 카드)



입문 예제 스튜디오에서 **07 손 찾기**를 실행한 화면. 손 관절 21곳에 점이 찍히고, 오른쪽 단추가 그 예제의 키입니다
(screen_startai_hand.png · 손 그림은 AI로 만든 그림)

- 1 맨 위 **프로그램** 목록에서 예제를 고르고 **▶ 시작** 을 누릅니다.
- 2 카메라 영상은 왼쪽, 그 예제의 키는 오른쪽 **한글 단추**(키보드로 눌러도 됩니다), 프로그램이 찍는 글은 오른쪽 **출력 칸**에 나옵니다.
- 3 프로그램이 무엇을 물으면(포트 번호, y/n 등) 단추 아래 **노란 입력 칸**에 답합니다. 끝내려면 **■ 끝내기** .

(다) 8단계 지도

단계	배선 핵심	실행하면
1 LED	LED 10개 D5~D9 · A0~A4 (+220Ω)	차례로 · 전체 · 웨이브 · 이진 카운터 0~15
2 도트 매트릭스	DIN D11 · CS D10 · CLK D13	8×8 점으로 숫자 · 무늬
3 네오픽셀	웹캠 + WS2812B	엄지 · 검지 거리만큼 무지개 LED
4 서보	서보 신호 D9	0도 ↔ 180도 왕복
5 릴레이	IN D4	모스부호 SOS 등 10가지

6 티처블 머신	웹캠	내 사진으로 학습한 AI가 분류
7 손 찾기	웹캠만	손 관절 21개를 점 · 선으로
8 통합	LED D8~D12 + 서보 + 웹캠	손 제스처 → 서보 · LED

예제는 시작할 때 포트를 찾고 **약 3초 준비**한 뒤 움직입니다. 그동안 안 움직여도 정상입니다.

(라) 6단계 티처블 머신 — 코딩 없이 나만의 AI

- 1 크롬으로 teachablemachine.withgoogle.com → 시작하기 → **이미지 프로젝트** → **표준 이미지 모델**.
- 2 클래스 이름을 정하고(예: 주먹 / 보) 웹캠으로 **한 클래스당 60장 이상**, 각도와 거리를 바꿔 가며 찍습니다.
- 3 **모델 학습**(탭을 닫지 않습니다) → **모델 내보내기** → **Tensorflow** → **Keras** → **다운로드**.
- 4 압축을 풀어 나온 [keras_model.h5](#) 와 [labels.txt](#) 를 2. [Start_AI\examples\06_teachable_machine\models\](#) 에 넣고 **06 티처블 머신 기본**을 실행합니다.

잘 안 맞으면

클래스는 2개로 작게, 배경은 단순하게, 조명은 밝게 하세요. 안 맞는 이유의 대부분은 "사진이 적거나 전부 같은 각도"입니다.

안 될 때

증상	원인	해결
No serial ports found	케이블 · 드라이버	데이터 케이블 확인, 장치 관리자에서 COM 확인
Check ... StandardFirmata	펌웨어를 안 올렸다	(가)를 다시 한다
keras_model.h5 없음	모델 파일 위치	(라) 4번의 폴더 확인
카메라가 안 켜짐	다른 프로그램이 쓰는 중	화상회의 · 카메라 앱을 닫고 다시

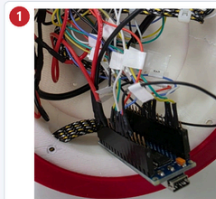
6축 로봇팔 — 미니 로봇팔로 조종

시간 조립 빠고 1차시 · 이번에는 일 미니 로봇팔(마스터)의 레버를 돌리면 큰 로봇팔(팔로워)의 서보 6개가 똑같이 따라 움직이게 한다.

미니 로봇팔의 관절마다 각도를 재는 **가변저항(포텐쇼미터)**이 들어 있어, 그 각도를 큰 로봇팔의 모터에 그대로 전달합니다. PC 없이 아두이노 나노 혼자 동작합니다. 로봇 농구에서 사람이 조종하는 쪽이 바로 이 로봇팔입니다.

로봇팔·미니로봇팔 회로도 키포인트

■ 제작 키포인트



신호선 구분

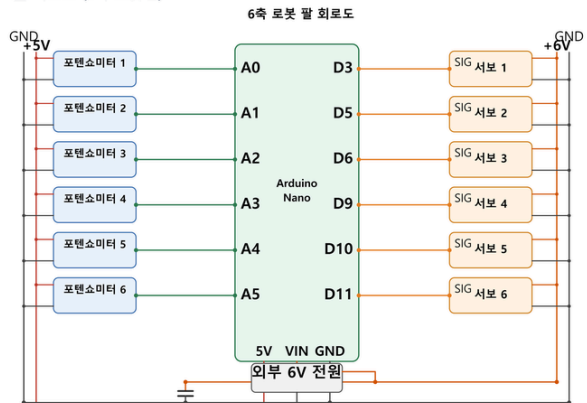
신호선을 구분하기 위하여 색을 달리하고, 번호 라벨을 붙이는 것을 권장한다.



전원 안정화 콘덴서

서보모터 과부하 시 시스템이 다운되는 현상을 방지하기 위해 2200µF·10V 콘덴서를 장착하는 것을 권장한다.

■ 회로도 (6축 로봇팔)



로봇팔 제어 보드 · 모터 배선 (photo_arm_wiring.png)

부분	연결
보드	Arduino Nano (시리얼 9600bps)
마스터 (입력)	포텐쇼미터 6개 → A0~A5
팔로워 (출력)	서보 6개 → D3 · D5 · D6 · D9 · D10 · D11

전원 주의

서보 6개와 나노는 6V를 함께 씁니다. 그러나 **포텐쇼미터 전원만은 반드시 나노의 5V 핀에서 받습니다** (6V를 바로 넣으면 아날로그 입력이 망가집니다). 전원 입구에는 2000µF 캐패시터를 달아 서보가 한꺼번에 움직일 때의 전압 출렁임을 막습니다.

(가) 펌웨어 올리기

Robot\Program\1. arduino\6-Axis Robot 폴더를 열고, 주소창에 powershell 을 입력해 그 폴더에서 명령 창을 엽니다. 아래를 붙여 넣습니다.

```
& "C:\Users\user\.platformio\penv\Scripts\platformio.exe" run -e nano -t upload
```

SUCCESS가 뜨면 성공입니다. avrdude 오류로 실패하면 `-e nano` 대신 `-e nano_new` 로 다시 합니다.

(나) 처음 한 번 — 레버 범위 맞추기(캘리브레이션)

```
& "C:\Users\user\.platformio\penv\Scripts\platformio.exe" device monitor -b 9600
```

- 1 부팅하면 `Press ENTER within 3s to calibrate.` 가 뜹니다. 3초 안에 `Enter`.
- 2 안내된 레버를 **최소에서 최대까지 끝까지** 돌립니다. 숫자가 더 안 변하면 `Enter` → 다음 채널.
- 3 6채널을 마치면 `Saved to EEPROM`. 다음부터는 바로 씁니다. 끝내려면 `Ctrl` + `C`.

성공

레버를 돌릴 때 서보 6개가 부드럽게 따라오면 성공입니다.

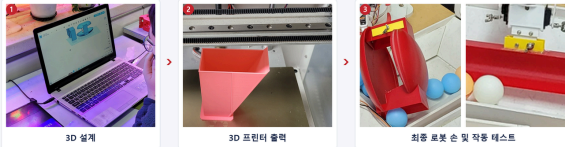
안 될 때

증상	원인	해결
글자가 깨진다	속도가 다르다	모니터를 9600으로
한 축만 안 움직인다	포텐쇼미터 배선	그 채널 배선 · 5V 확인
부팅하자마자 팔이 튕다	범위 값이 레버와 안 맞음	캘리브레이션을 다시
서보가 다 같이 떴다	전원 부족	캐패시터 · 6V 전원 용량 확인

나만의 그리퍼 만들기 — 3D 설계와 출력

시간 설계 1차시 + 출력 + 조립 · 시험 1차시 · **이번에 하는 일** 로봇팔 끝에서 탁구공을 집는 그리퍼(로봇 손)를 직접 설계 · 출력해 단다.

로봇 손(그리퍼) 제작 과정



그리퍼 3D 모델링과 출력물 (photo_gripper_design.png)

동작 검증 및 작동 시험

카메라-로봇 좌표 조정 후 실제 환경에서 반복 동작과 픽업-투입 정확도를 점검한다.



학생이 만든 그리퍼 장착 · 시험 (photo_gripper_test.png)

(가) 설계 — TinkerCAD

1. tinkercad.com 에 로그인 → 새 디자인.
2. 상자 · 원기둥을 끌어와 **mm** 단위로 크기 · 위치를 맞춥니다.
3. 빼낼 자리에 도형을 겹치고 **구멍(Hole)**으로 바꾼 뒤 함께 골라 **그룹화**(**Ctrl** + **G**). 서보 축 구멍 · 나사 구멍을 이렇게 팝니다.
4. 탁구공 지름 **약 40mm**가 들어가게 손가락 벌어진을 잡습니다.

(나) 출력 — STL → Cura → 프린트

TinkerCAD에서 **내보내기** → **.STL**. Cura에서 PLA, 레이어 0.2mm, 채움 20~40%, 공중 부분은 서포트, 바닥은 brim으로 슬라이스해 출력합니다.

(다) 달고 시험한다

그리퍼를 달고 탁구공을 **반복해서** 집어 봅니다. 놓치면 "왜?(벌어진 · 마찰 · 높이)"를 분석해 **다시 설계 · 출력**합니다. 이 개선의 순환이 이 활동의 핵심입니다. 경기에서 지면 손을 고쳐 다시 도전합니다.

더 잘 잡으려면	어떻게
미끄럼 방지	손가락 안쪽에 홈 · 돌기, 또는 고무 패드
삼각 파지	손가락을 120도 간격 3개로
서보 축 구멍이 안 맞음	구멍 지름을 0.2~0.5mm 키워 다시 출력

탁구공 로봇팔 — 키보드 · 손 · AI 자동

시간 2차시 · 이번에 하는 일 같은 로봇팔을 키보드 → 손 → AI 자동 순서로 움직여 보고, AI가 스스로 탁구공을 집어 골대로 옮기게 한다.

Robot\Program\4. auto_pingpong_system - stepping → 탁구공스튜디오_시작.bat (서보 버전은 5. auto_pingpong_system_servo 의 같은 이름)

창을 열면 아두이노가 하나 보일 때 스스로 연결하고, 연결되면 로봇이 홈으로 움직입니다(30초쯤). 위쪽 줄의 상태가 바뀔 때까지 기다리세요. 탭 넷을 차례로 씁니다.

(가) ① 키보드 조종



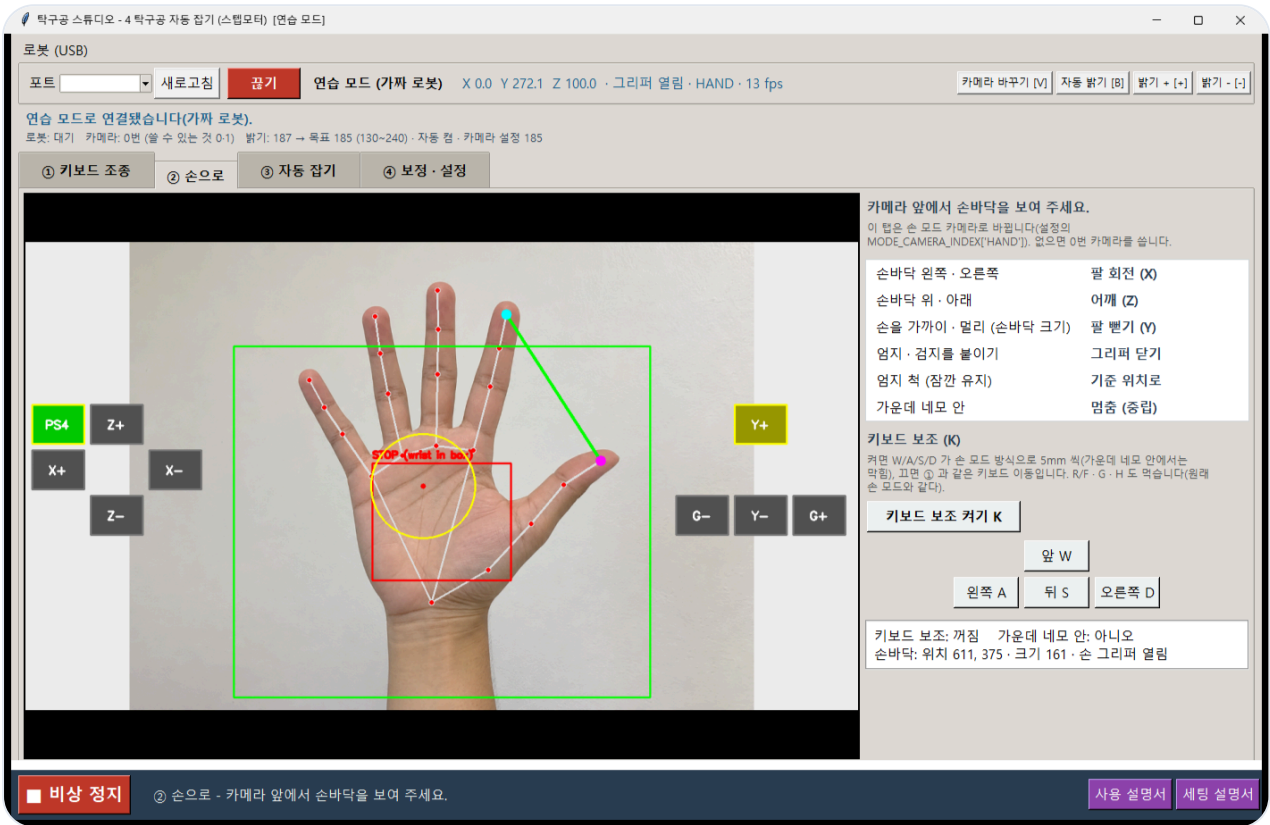
① 키보드 조종. 오른쪽 단추나 키보드로 팔을 움직입니다 (screen_pingpong_keyboard.png)

키	하는 일
W / S	앞 / 뒤
A / D	왼쪽 / 오른쪽
R / F	위 / 아래 (방향키 · PgUp/PgDn 도 된다)

G	그리퍼 열기 · 닫기
H	홈(처음 자리)
U I O · Z X C	속도 · 한 번에 움직이는 단위

한글 입력 상태에서도 됩니다. 칸(포트 등)에 커서가 있으면 키가 로봇으로 안 갑니다 — **Esc** 를 누르거나 영상을 한 번 누르세요.

(나) ② 손으로



② 손으로. 노트북 카메라가 손을 찾고, 오른쪽 표처럼 손 위치 · 크기 · 손가락으로 팔을 움직입니다 (screen_pingpong_hand.png · 손 그림은 AI로 만든 그림)

손	로봇팔
손바닥 왼쪽 · 오른쪽	팔 회전
손바닥 위 · 아래	어깨
손을 가까이 · 멀리 (손바닥 크기)	팔 뻗기
엄지 · 검지를 붙이기	그리퍼 닫기

엄지 척 (잠깐 유지)	기준 위치
가운데 네모 안	멈춤

손 조종은 노트북 카메라, 나머지는 천장(외장) 카메라를 씁니다. 탭을 옮기면 카메라도 저절로 바뀝니다.

(다) ③ 자동 잡기 — AI가 스스로



③ 자동 잡기. 초록 작업 박스 안의 25개 점과 찾은 공이 보입니다. [▶ 자동 잡기 시작]을 누르면 공을 차례로 집어 골대로 옮깁니다 (screen_pingpong_auto.png)

- 1 작업판의 초록 박스 안에 공을 1~5개 놓습니다.
- 2 ▶ 자동 잡기 시작 (또는 Space). 로봇이 접근 → 잡기 → 홈 → 골대 → 놓기를 되풀이합니다.
- 3 잡고 올라온 뒤 카메라로 공이 정말 사라졌는지 확인합니다. 놓쳤으면 골대 대신 다시 잡으려 갑니다(잡기 확인).

오른쪽 판에서 보이는 공 개수 · 목표 점 · 성공률을 봅니다. 밝기는 위 줄의 단추나 + - B (자동).

(라) ㉔ 보정 · 설정 — 잘 안 잡힐 때

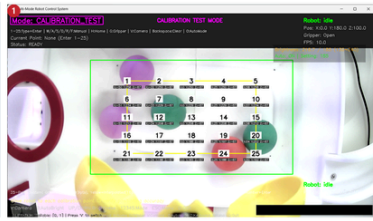


㉔ 보정 · 설정. 작업 박스 · 픽업 옵션 · 점 이동 · YOLO 위치를 맞추니다 (screen_pingpong_calib.png)

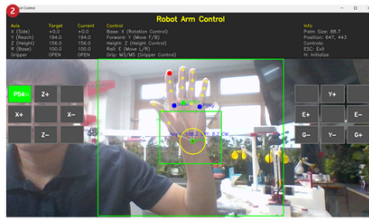
작은 탭	언제	어떻게
㉔ 작업 박스	카메라나 작업판을 옮겼을 때	영상에서 작업판 모서리를 차례로 클릭
㉔ 픽업 옵션	공 옆을 집거나 헛집을 때	숫자패드 8/2 앞뒤 · 4/6 좌우 · 7/1 높이 · 5 저장
㉔ 점 이동	25개 점이 맞는지 볼 때	번호(1~26) 입력 후 이동. 26은 골대
㉔ YOLO 위치 조정	공 표시가 실제와 어긋날 때	W/S/A/D 로 옮기고 O 저장

보정이 왜 필요할까

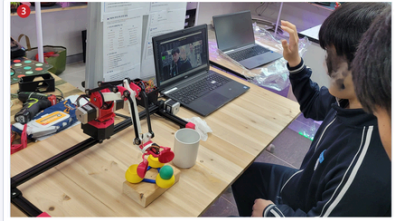
카메라가 본 위치(화면의 픽셀)를 로봇이 움직일 좌표(mm)로 바꿔야 합니다. 그래서 작업판 여러 지점에서 카메라 좌표와 로봇 좌표를 미리 맞춰 둡니다.



카메라-로봇 좌표 매핑 및 그리퍼 높이 보정



손목을 화면의 중앙에 위치시키고 손의 높이를 조절하여 움직이지 않는 상태를 찾은 뒤, 좌우·상하·앞뒤로 움직이며 로봇팔의 이동을 점검한다.



구글의 미디어파이프를 활용한 손인식 로봇팔 조종

카메라 – 로봇 좌표 맞추기와 그리퍼 높이 보정 (photo_calibration.png)

관찰하고 생각하기

1. 키보드 · 손 · AI 자동 중 어느 방법이 가장 빠르고, 어느 방법이 가장 정확합니까?
2. 공이 작업 박스 밖에 있으면 AI는 왜 잡지 않습니까?
3. 스텝 버전(4번)과 서보 버전(5번)으로 같은 공 배치를 잡게 하면 무엇이 다른니까?

안 될 때

증상	원인	해결
연결이 안 된다	보드가 여럿 · 포트가 틀림	포트를 직접 고르고 연결
헛집는다	잡는 높이 · 위치	④ ④ 픽업 옵션에서 높이 · 앞뒤를 조금씩
공을 못 찾는다	너무 밝거나 어둡다 · 배경	밝기 조정, 흰 판 위에 색 공
모터가 찌그덕(탈조)	너무 빠르다	속도를 낮춘다
키가 안 먹는다	입력 칸에 커서	Esc 또는 영상을 한 번 클릭

모방학습 — 로봇에게 잡는 법을 가르치기

시간 2~3차시 · 이번에는 하는 일 사람이 시범을 보이면 로봇이 배워서 처음 보는 자리의 공도 잡게 한다. 가르치는 방법 다섯 가지를 비교한다.

Robot\Program\10. imitation_learning_robot_servo → 모방학습스튜디오_시작.bat (스텝 버전은 10. imitation_learning_robot - stepping)

탭 하나가 가르치는 방법 하나입니다. 공은 작업판의 **노란 박스** 안에 둡니다.

탭	핵심 조작	배우는 것
② 키보드 티칭	W/S/A/D/R/F 이동 · G 그리퍼 ·  녹화 시작/끝 · P 재생	녹화는 일반화가 안 된다 — 공을 옮기면 빛나간다
① 손 시연	손을 보이면 로봇이 따라온다 · R 녹화(3초 뒤) · S 저장 · P 재생	시연 기록의 한계
④ 위치 티칭 (KNN)	Y 공 고르기 → 이동 → T 점 저장을 6~10곳 → Space 잡기	가까운 점 3개의 가중 평균(보간)
③ ML 모방 (회귀)	Y 공 고르기 → 이동 → T 가르치기 6번 이상 → M 학습 → Space	식 하나로 학습(밖으로도 늘려 짐작)
⑤ YOLO 자동 잡기	Space 한 번 · A 연속	배운 것을 실전에

가르치는 순서 제안

②로 "그냥 녹화"의 한계를 먼저 겪고 → ④로 점을 가르치고 → ③으로 학습하고 → ⑤로 자동. **실패를 먼저 겪어야** KNN과 회귀가 왜 필요한지 몸으로 압니다.

(가) ② 키보드 티칭 — 녹화와 재생



② 키보드 티칭 (screen_learning_keyboard.png)

Enter 로 녹화를 시작하고 키보드로 공을 집어 골대에 넣은 뒤 다시 **Enter** 로 저장합니다. **P** 로 재생하면 로봇이 그대로 따라 합니다. 이제 공을 다른 자리에 놓고 다시 재생해 보세요. 빗나갑니다. 왜일까요?

(나) ① 손 시연



① 손 시연. 손을 보이면 로봇이 바로 따라옵니다 (screen_learning_hand.png · 손 그림은 AI로 만든 그림)

(다) ④ 위치 티칭 (KNN) 과 ③ ML 모방 (회귀)



④ 위치 티칭. 작업판 여러 곳에서 "공 자리 → 잡는 자세"를 저장합니다 (screen_learning_points.png)

- 1 Y 로 공을 고르고, 키보드로 팔을 공 위로 옮겨 잡을 자세를 만든 뒤 T 로 저장합니다.
- 2 공을 옮겨 가며 격자처럼 고르게 6~10곳을 가르칩니다.
- 3 새 자리에 공을 놓고 Space . 가르친 점 중 가까운 셋을 섞어 자세를 짐작해 잡습니다.



③ ML 모방. 같은 방법으로 6번 이상 가르친 뒤 [④ 학습 M]을 누르면 식을 만들어 오차를 보여 줍니다 (screen_learning_ml.png)

(라) KNN과 회귀 겨루기 — 데이터 과학 실험

1. ④에서 3×3 격자 9점을 가르칩니다.
2. **영역 안** — 가르친 점 사이에 공을 놓고 10번 잡기, 성공 횟수를 적습니다.
3. ③에서 같은 점으로 학습(M)한 뒤 같은 자리에서 10번.
4. **영역 밖** — 가르치지 않은 가장자리에서 두 방식을 비교합니다.

방식	영역 안 성공 (/10)	영역 밖 성공 (/10)
④ 위치 티칭 (KNN)		
③ ML 모방 (회귀)		

예상 — 안쪽은 KNN이 정확하고, 바깥은 회귀가 덜 무너집니다. 직접 확인해 보세요.

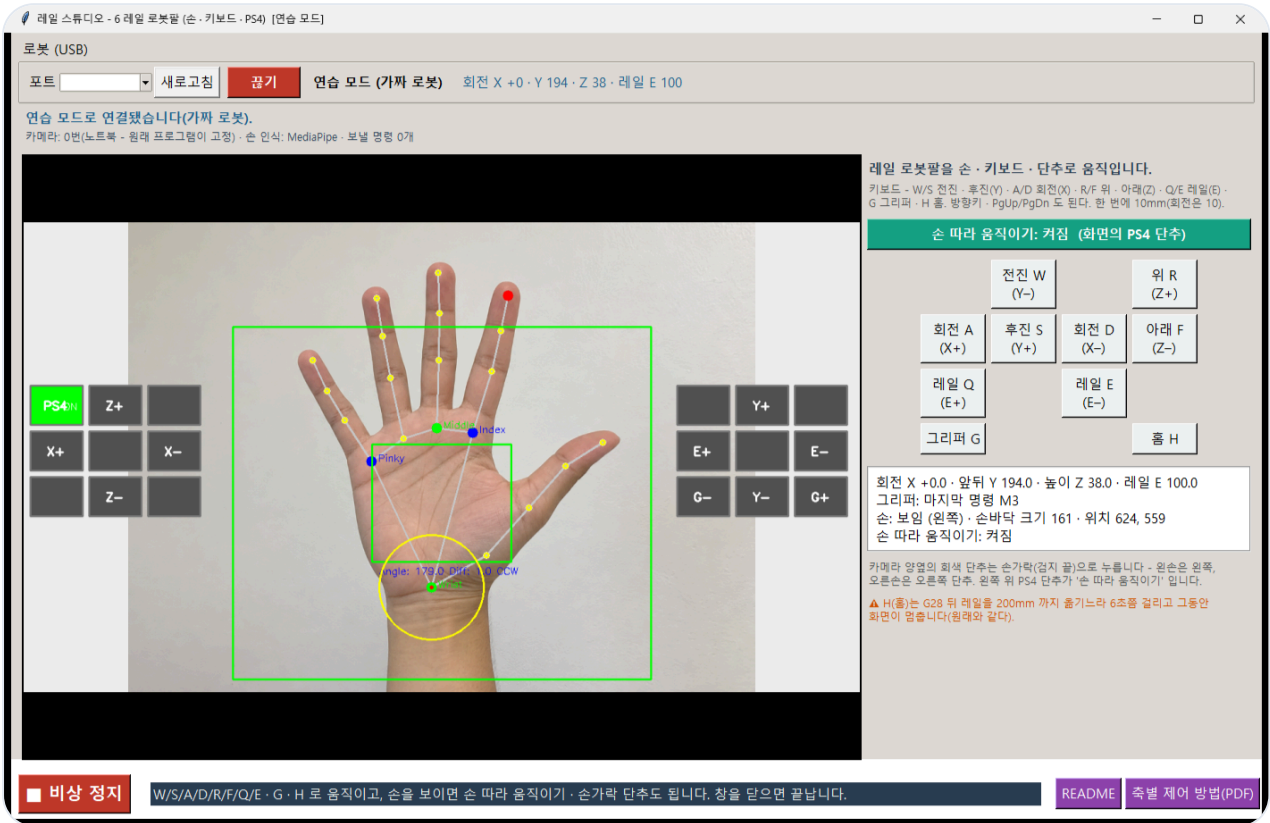
안 될 때

증상	원인	해결
공이 안 잡힌다	가르친 점이 적다 · 영역 밖	점을 더 촘촘히, 공은 노란 박스 안
재생이 빗나간다 (㉔)	녹화는 그 자리에서만 맞다	정상이다. ㉓ · ㉔로 넘어간다
Space가 안 된다 (㉓ · ㉔)	공을 안 골랐다	 로 공부터 고른다
탁구공 스튜디오와 함께 안 된다	같은 COM 포트	하나만 켜다

레일 로봇팔 — 넓어진 작업 공간

시간 1차시 · 이번에는 하는 일 로봇팔에 레일(좌우 0~400mm)을 더한 4축 로봇을 손 · 키보드 · 화면 단추로 움직인다.

Robot\Program\6_Robot_Stepping_Rail → 레일스튜디오_시작.bat



레일 스튜디오. 손을 보이면 손을 따라 움직이고, 카메라 양옆의 회색 단추를 손가락으로 "누를" 수도 있습니다 (screen_rail.png · 손 그림은 AI로 만든 그림)

키 · 단추	하는 일
W / S	전진 / 후진 (Y)
A / D	회전 (X)
R / F	위 / 아래 (Z)
Q / E	레일 (E)
G · H	그리퍼 · 홈 (홈은 6초쯤 화면이 멈춘다)
손 따라 움직이기	손바닥이 가운데 네모 밖으로 가면 그쪽으로 팔이 움직인다

■ 비상 정지

밀린 명령을 버리고 모터를 끈다

PS4 게임패드도 됩니다. **프로그램을 켜기 전에** USB로 꽂거나 블루투스로 연결합니다(블루투스는 SHARE + PS 버튼을 3초 눌러 깜빡이게 한 뒤 Windows 설정 → Bluetooth → 장치 추가 → Wireless Controller).

레일 로봇에게 가르치기

9. `education_rail_robot` 의 **모방학습스튜디오_시작.bat**은 위 모방학습을 레일로 넓힌 것입니다.

화면의 X가 실제로는 **레일 위치**라는 것만 다릅니다. 레일 양 끝에 공을 두고 가르치면 로봇 하나가 책상 하나 폭을 다룹니다.

그림 그리는 로봇 — 드로잉

시간 2차시 · 이번에 하는 일 사진을 로봇이 그릴 수 있는 선(G-code)으로 바꾸고, 펜을 든 로봇팔이 종이에 그리게 한다.

사진을 넣으면 AI가 얼굴이나 사물의 윤곽선을 뽑아내고, 로봇팔이 그 선을 따라 그림을 그립니다. 다섯 단계입니다 — 사진 → 화풍 바꾸기(선택) → 선으로 바꾸기 → 미리보기 → 그리기.

(가) 그리기 좋은 그림 만들기 (선택)

일반 사진은 명암 · 배경이 복잡해 선으로 바꾸면 지저분합니다. 그림 생성 AI에 `이 사진을 애니메이션 스타일로 그려줘. 윤곽과 이목구비는 유지하고 배경은 단순하게.` 처럼 부탁하면 윤곽이 또렷해집니다. 받은 그림을 `Robot\Program\7. drawing` 폴더에 `photo.png` 라는 이름으로 넣습니다.

개인정보

친구 얼굴을 쓸 때는 먼저 허락을 받고, 만든 사진 · 그림을 함부로 올리지 않습니다.

(나) 사진 → 선(G-code)

`7. drawing` 폴더에서 명령 창을 열고(주소창에 `powershell`) 붙여 넣습니다.

```
..\conda\python.exe face_to_gcode.py photo.png --quality HIGH
```

품질	점 개수	용도
LOW	약 100	빠른 시험
HIGH	약 500	권장
ULTRA	약 1000	최고 품질

눈 · 코 · 입을 빼려면 `--no-eyes --no-nose --no-mouth` , 크기는 `--work-width 150 --work-height 150` (mm). 결과는 `face_output.gcode` 입니다.

(다) 미리보기 → 연결 → 그리기

```
..\conda\python.exe gcode_preview.py face_output.gcode
```

```
..\conda\python.exe drawing_control_gui.py
```

- 1 미리보기에서 경로가 빨간 점선(작업 영역) 안에 드는지 봅니다.
- 2 제어 화면에서 포트를 고르고 **연결**(홈으로 움직입니다).
- 3 그리퍼에 펜을 끼우고 종이를 고정한 뒤, 이동 단추로 시작 위치에 가서 **Z**-로 펜이 종이에 살짝 닿게 하고 **원점 설정**.
- 4 G-code 파일을 고르고 속도 50~75%로 **실행**. **Space** 일시정지, **Esc** 중지.

증상	해결
펜이 긁거나 안 닿는다	Z-로 높이를 다시 맞추고 원점 설정
그림이 종이 밖으로 나간다	미리보기 확인, 시작점 · 원점 다시
선이 지저분하다	(가)의 화풍 바꾸기, 또는 품질을 낮추고 부위 빼기

로봇 농구 경기 — 내 그리퍼의 실전

시간 1차시 + 연습 · 이번에 하는 일 직접 만든 그리퍼를 단 로봇팔로 탁구공을 골대에 넣는 경기를 한다. 경기장이 자동으로 점수를 센다.

로봇 농구 경기장 제작 키포인트

제작 키포인트

1



기본 조립 (3D 출력물)

초기에는 3D 출력물만으로 조립하여 사용하기를 권장하며, 기본 기능 구현에는 무리가 없다.

2



경기장 기울기

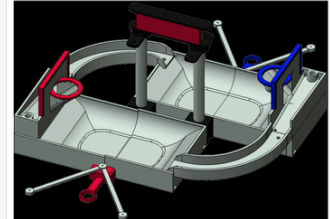
공이 상대방 경기장으로 자연스럽게 이동할 수 있도록 적절한 기울기를 고려하여 설치한다.

3



골대 경첩(장석) 연결

농구 골대를 장석(경첩)으로 연결하여 로봇팔과 충돌할 경우 발생하는 파손을 방지한다.



■ 완성 모델 (3D 설계)

학생이 창의적으로 제작한 로봇손을 장착한 로봇팔과 인공지능 로봇팔을 손쉽게 교체·장착할 수 있도록 탈착형 구조로 설계하였다.

경기 시작은 물리적 스위치와 스마트폰 앱이 동일하게 연동되어 작동한다.

4



카메라 배치

카메라는 한쪽 방향에만 설치하되, 경기장 중앙을 향하도록 배치한다.

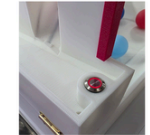
5



거리측정 센서

센서는 위쪽을 향하도록 설치한다. 고정 나사를 과도하게 조이면 센서가 손상될 수 있으므로 주의한다.

6



시작 스위치

경기 시작 스위치는 모멘터리 방식을 사용하며, 규격은 5V-12파이 제품을 적용한다.

로봇 농구 경기장 — 기울기 · 골대 · 카메라 · 거리측정 센서 · 시작 스위치 (photo_stage_keypoints.png)

(가) 경기장 켜고 접속한다

1

경기장 전원을 켭니다. 켜질 때 골대 거리센서가 스스로 기준을 잡으므로 골대에 공이 없어야 합니다.

2

폰이나 PC를 경기장 Wi-Fi **Fab_Lab** (비밀번호 **12345678**)에 연결합니다.

3

크롬으로 <http://192.168.4.1> 을 엽니다. 점수판과 시작 단추가 나옵니다.

(나) 경기 규칙

- 양 팀 탁구공 **5개씩, 90초**.
- 공을 골대에 넣으면 경사로를 타고 **상대편으로 넘어갑니다**.
- 90초가 지나거나 한쪽 공이 다 없어지면 끝. **남은 공이 적은 쪽이 이깁니다**.

(다) 대전 방식

방식	무엇
사람 대 사람	모둠마다 만든 그리퍼를 단 6축 로봇팔(미니 로봇팔로 조종)로 대결

사람 대 AI

사람이 조종하는 로봇팔 대 스스로 잡는 AI 로봇팔(탁구공 로봇팔 ③ 자동)

진 경기는 "왜 놓쳤나"를 분석해 그리퍼를 다시 설계합니다. **경기가 곧 성찰 도구**입니다.

증상	원인	해결
점수판 화면이 안 열린다	Wi-Fi	<code>Fab_Lab</code> 에 다시 연결 → 192.168.4.1
골을 넣었는데 점수가 안 오른다	센서가 못 봤다	공이 센서 사이를 지났는지, 켄 때 골대에 공이 있었는지 확인

컨베이어 벨트 — 공급 자동화

시간 1차시 · 이번에 하는 일 물체를 자동으로 떨어뜨려 벨트로 나르고, 로봇이 가져갈 시간을 기다리는 공급 장치를 다룬다.

컨베이어 벨트 제작 키포인트

1 제작 키포인트

1 벨트 장착 조절
벨트 위 공의 무게를 견딜 수 있도록 적절한 장력 조절이 필요하며, 롤러를 고정하여 장력을 조절할 수 있다.

2 스텝모터·롤러 연결
스텝모터와 롤러를 연결할 때, 벨트를 장력한 상태에서 축에 롤러를 끼우면 손쉽게 조립할 수 있다.

3 롤러 직각유격
외관이 정렬되도록 적절한 유격을 주고, 프레임과 직각을 이루게 하여 벨트의 이동을 방지한다.

4 벨트 봉제 고정
벨트는 장력 장치보다 실패 빠느릴수록 고정해야 하며, 그레와 장력으로 인한 이동을 줄여줄 수 있다.

5 서보모터 각도 배선
서보모터가 공을 고정했다 떨어뜨리도록 각도 조절에 유의하고, 배선이 벨트에 간섭하지 않도록 정리한다.

6 작동 옵션 설정
속도, 공의 개수, 투입 간격 등 다양한 옵션이 있다. 처음에는 속도를 느리게 설정하여 작동 상태를 확인한다.

■ 완성 모델 (3D 설계)
미디아라이프(MedupPgo)를 활용하여 손쉽게 3D로 제작하는 완성형, 그리고 서보를 활용한 속도·방향·작업을 수행할 때 거점 코드를 적기 위해 사용한다.
벨트를 적지 않게 할 때에는 장력 조절과 함께 벨트가 앞으로 흐르지 않도록 롤러와 수직을 유지하는 것이 중요하다.

컨베이어 벨트 회로도 키포인트

1 작동 투하·전달 시퀀스
서보로 물체를 투하한 뒤, 물체 거리만큼 벨트를 이동시키고, 로봇팔이 잡을 때까지 대기한다. 공이 떨어지지 않으면 벨트 끝으로 계속하여, 이 1사이클을 설정한 횟수만큼 반복한다.

2 수동 벨트 구동
버튼으로 벨트를 즉시 시작·정지할 수 있으며, 비동전 방식으로 구동되어 다른 제어와 독립적으로 작동한다.

3 실시간 속도 조절
좌우 버튼으로 벨트 이동 속도를 0~100% 범위에서 실시간으로 조절한다.

4 설정 모드 (4방향)
이동 속도 이동 거리·대기 시간·반복 횟수를 설정하며, 설정값은 EEPROM에 저장되어 전원을 꺼도 유지된다.

5 로봇팔 연동·LCD 표시
대기 중 수거 신호를 받으면 배율을 건너뛰어 로봇팔과 연동된다. LCD에는 대기 시간·인행률·사이클이 실시간으로 표시된다.

컨베이어 제어부 배선 (photo_conveyor_wiring.png)

Arduino Uno + 1602 LCD 키패드(버튼 6개) + 스텝모터 드라이버(TMC2208) + 투하 서보로 되어 있습니다. 펌웨어가 올라가 있으면 **PC 없이 LCD 버튼만으로** 조작합니다.

버튼	하는 일
Up	자동 시작 (투하 → 벨트 이동 → 대기 → 안 가져가면 버림 → 반복)
Down	벨트 수동 시작 · 정지
Right / Left	속도 올리기 / 내리기. 대기 중 Right = "로봇이 가져갔다" 신호
Select	설정(속도 · 이동거리 · 대기시간 · 반복횟수). Select로 저장

펌웨어를 다시 올릴 때는 우노를 USB로 꽂고 **8. conveyor** 폴더에서 명령 창을 열어 붙여 넣습니다.

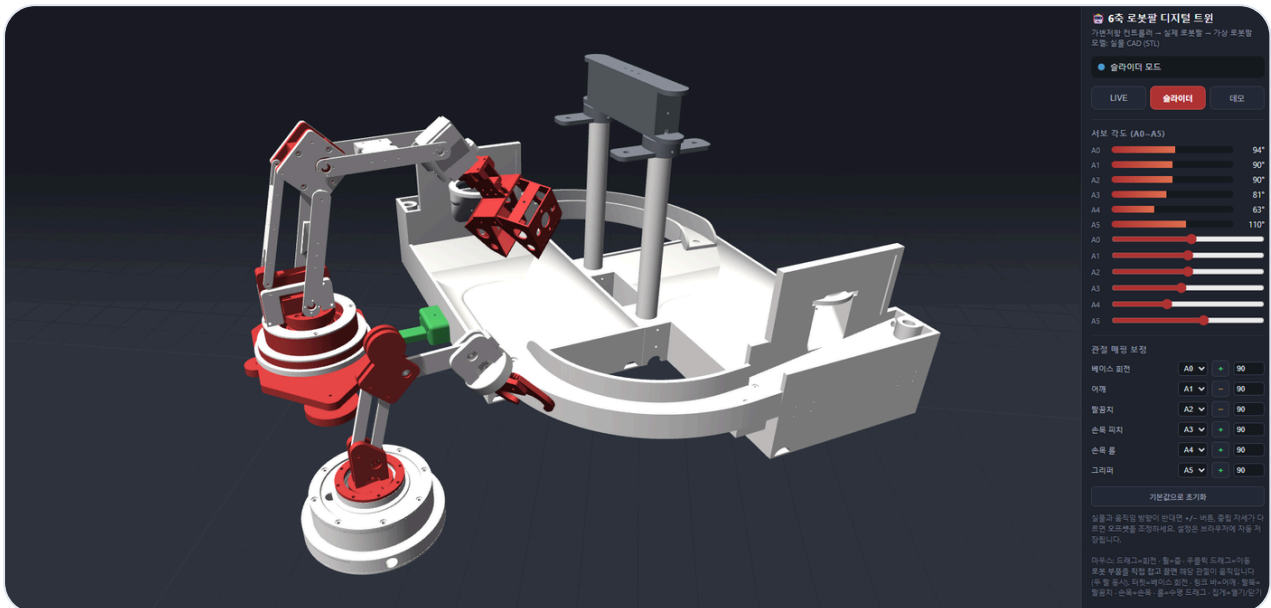
```
& "C:\Users\user\.platformio\penv\Scripts\platformio.exe" run -e uno -t upload
```

창의 종합 — 풀 자동화 라인

컨베이어(공급) → 로봇팔(잡기 · 운반) → 경기장(투입 · 채점)을 하나로 이으면 무인 생산 라인이 됩니다. 어디서 기다리고 어떤 신호로 넘길지 설계해 보세요.

디지털 트윈 — 화면 속 쌍둥이 로봇팔

시간 30분 · 이번에 하는 일 미니 로봇팔을 손으로 움직이면 화면 속 3D 로봇팔이 실시간으로 똑같이 움직이게 한다.



6축 로봇팔 디지털 트윈. 실물 CAD 그대로의 3D 로봇팔과 관절 각도 막대 (screen_twin.png)

Robot\Program\11. digital_twin → 디지털트윈_실행.bat (경기장까지 보려면 디지털트윈_경기장_실행.bat)

검은 창(서버)이 뜨고 브라우저가 열립니다(<http://127.0.0.1:8137>). 미니 로봇팔이 USB로 꽂혀 있으면 스스로 잡습니다.

모드	무엇	언제
LIVE	미니 로봇팔의 각도를 받아 실시간으로 재현	실물을 연결했을 때
슬라이더	화면 막대로 관절을 직접 움직임	관절 공부 · 실물 없이
데모	관절이 스스로 시연 동작	전시 · 소개

화면 조작 — 끌기 = 회전, 휠 = 확대, 오른쪽 끌기 = 이동. 3D 부품을 마우스로 잡고 끌면 그 관절이 움직입니다. 실물과 **반대로** 움직이면 오른쪽 보정 패널의 +/-, 가운데 자세가 어긋나면 **오프셋**을 바꿉니다.

실물 없이 해 보기

11. `digital_twin` 폴더에서 명령 창을 열고 `.\디지털트윈_실행.bat --no-serial` 로 실행하면 슬라이더 · 데모로 관절 6개를 마음껏 움직여 볼 수 있습니다.

안 될 때 — 증상별 해결 모음

증상	원인	해결
"찾을 수 없습니다"	다른 폴더에서 명령을 쳤다	명령 창의 PS ...> 경로가 그 프로젝트 폴더인지 본다
보드 연결이 안 된다	케이블 · 드라이버 · 포트	데이터 케이블, 장치 관리자의 COM 번호, 포트 직접 고르기
두 프로그램이 다 안 된다	같은 로봇(같은 COM)을 함께 켜다	하나만 켜다. 4번과 10번(스텝)은 같은 로봇
카메라가 안 켜진다	다른 프로그램이 쓰는 중	화상회의 · 카메라 앱 닫기
공을 못 찾는다	밝기 · 배경	밝기 조정(B 자동), 흰 판 위에 색 공
서보가 떨리고 리셋된다	전원 부족	서보는 외부 전원으로
비상 정지 뒤 안 움직인다	모터가 꺼져 있다	끊기 뒤 연결
창 맨 아래 줄에 빨간 글	프로그램 오류	그 글을 그대로 선생님이나 AI에게 보여 준다

부록 A. AI에게 물어보기

막혔을 때는 상황을 다섯 칸으로 적어 물어보면 답이 정확해집니다. 아래를 복사해 빈칸을 채우세요.

[환경] 윈도우 노트북, Fab-Lab 로봇 (몇 번 폴더), (스튜디오 이름과 탭)
[목표] (하고 싶은 것)
[상황] (화면에 나온 글자와 숫자를 그대로)
[시도] (이미 해 본 것)
[부탁] 원인을 찾는 순서와, 고친다면 어느 파일의 어느 값인지 알려 주세요. 전체 코드를 새로 써 주지는 마세요.

예시

[환경] 10. imitation_learning_robot_servo, 모방학습 스튜디오 ④ 위치 티칭
[목표] 가장 가까운 점 3개 대신 5개를 섞어 쓰면 어떻게 달라지는지 실험하고 싶다
[상황] 3x3 격자 9점을 가르쳤고 영역 안 성공률은 8/10 이다
[부탁] k=3 을 5로 바꾸면 무엇이 달라지는지, 어느 파일의 어느 값을 고치면 되는지 알려 주세요.

AI 답을 믿기 전에

1. 그 프로그램에 정말 그런 기능·값이 있는지 화면이나 파일에서 확인합니다.
2. 로봇 근처를 비우고, 느린 속도로 먼저 시험합니다.
3. 한 번에 한 곳만 바꿉니다.

부록 B. 더 보고 싶을 때

이 교재는 수업 핵심만 담았습니다. 더 깊이 하고 싶으면 아래 원래 문서를 보세요. 모두 `D:\data\study\AI\Fab-Lab\Robot\` 아래에 있습니다.

예전 교재 묶음(교재모음 · 종합 교재 · 배포 zip)은 `D:\data\study\AI\Fab-Lab\보관\예전교재_2026-10-13` 로 옮겼습니다. 프로그램 폴더 안의 README · 사용설명서는 그대로 있습니다.

알고 싶은 것	문서
11개 프로젝트 전체를 자세히 (개정 2판)	<code>D:\data\study\AI\Fab-Lab\보관\예전교재_2026-10-13\Robot\교재\로봇팔_교재_v2\로봇팔_교재_종합_v2.html</code>
스튜디오의 모든 단추 · 키	<code>Program\00_로봇스튜디오\README.md</code>
Start_AI 단계별 설명 · 원리 해설	<code>Program\2. Start_AI\examples\(\단계)\TUTORIAL.md</code> · <code>AI_GUIDE.md</code>
탁구공 로봇팔 설정값 (골대 위치 · 속도 · 감지 민감도)	<code>Program\4. auto_pingpong_system - stepping\USER_MANUAL.md</code> · <code>config_auto.py</code>
모방학습 모드 자세히	<code>Program\10. imitation_learning_robot_servo\사용설명서_모방학습.md</code>
레일 로봇 교육 6모드	<code>Program\9. education_rail_robot\사용설명서_레일로봇.md</code>
드로잉 옵션 전부	<code>Program\7. drawing\README_face_to_gcode.md</code>
디지털 트윈 운용	<code>Program\11. digital_twin\교재_디지털트윈_운용.md</code>
예전 교재 34권 전체 목록	<code>D:\data\study\AI\Fab-Lab\보관\예전교재_2026-10-13\Robot\교재모음\index.html</code>