

[한국기계연구원 동남권기계연구본부]

Claude Code 활용 교육

연구·시험 업무에 AI 에이전트 쓰기 · 오전 강의

2026. 10. 15. · 이철로 (지피타쿠) · 엑스워드(AXwith)

스레드 클코대장 지피타쿠

gptaku_ai

비즈니스 및 기업 강의 문의

✉ gptaku@axwith.com

▼ 클코단 오픈채팅방 링크

open.kakao.com/o/g7t4M0Fi +1 ▾

🔗 AI Threads

AI

ClaudeCode

+

팔로워 4.0만명 · 최근 조회수 247.9만회



이철로 지피타쿠

- 현) **AX with** — AX 컨설팅 & AI 교육 설계
- 현) **1인 개발자** — AI 기반 서비스 개발
- 전) **R&D 메카니컬 엔지니어 10년**
(GENTLE MONSTER, LG INNOTEK 등)

Claude Code 플러그인 모음 **gptaku-plugins**를 만들어 공개하고 있습니다. 2부에서 연구개발에 쓰는 조합을 보여드립니다

오늘 일정

오전에는 세미나, 오후에는 Claude Code 강의와 실습을 진행합니다

오전 · 세미나

1. 왜 써야 하나

채팅 AI와 AI 에이전트의 차이

2. 어떻게 돌아가나

에이전트 루프와 연구 워크스페이스

3. 어떻게 쓰나

연구·시험 업무 사례와 플러그인 조합

4. 장비·하드웨어에는

장비 연동, 피지컬 AI, Jev 동향

오후 · 강의와 실습

1. 처음 맡겨 보기

빈 폴더에서 자료 받기·그래프·검토

2. 내 워크스페이스 만들기

인터뷰와 내 자료 10개로 채웁니다

3. 주제 하나로 끝까지

논문, 공고 적합도, 기획서

4. 계속 쓰게 만들기

스킬, 기록, 예약 실행

오늘 가져가는 것 · 내 자료로 채운 연구 워크스페이스 한 벌과, 공고 적합도 판정·기획서 초안·업무 스킬

시작하며 · 1/3

1. 업무에 AI를 쓰고 계신가요?

시작하며 · 2/3

2. Claude Code를 써 보셨나요?

시작하며 · 3/3

3. Claude Code로 내 업무용 프로그램이나 스킬을 만들어 쓰고 계신가요?

1부

Claude Code란

무엇이고, 어떻게 동작하고, 왜 쓰며, 무엇을 할 수 있는지 봅니다

1부 · 무엇인가

Claude Code는 내 PC의 폴더에서 직접 일하는 AI 에이전트입니다

웹 채팅 AI

claude.ai · ChatGPT

이 측정 파일로 시간이력 그래프 그려 줘

📎 Waveforms Point05.tdms

아래 코드를 PC에서 실행하면 그래프를 그릴 수 있습니다.

```
from nptdms import TdmsFile
f = TdmsFile.read("../tdms")
...
```

실행 결과를 붙여 주시면 이어서 도와드릴게요.

답과 코드가 **대화창에** 남습니다. 실행과 저장은 내가 합니다.

Claude Code

Claude Code — demo

> 이 측정 파일로 시간이력 그래프 그려 줘

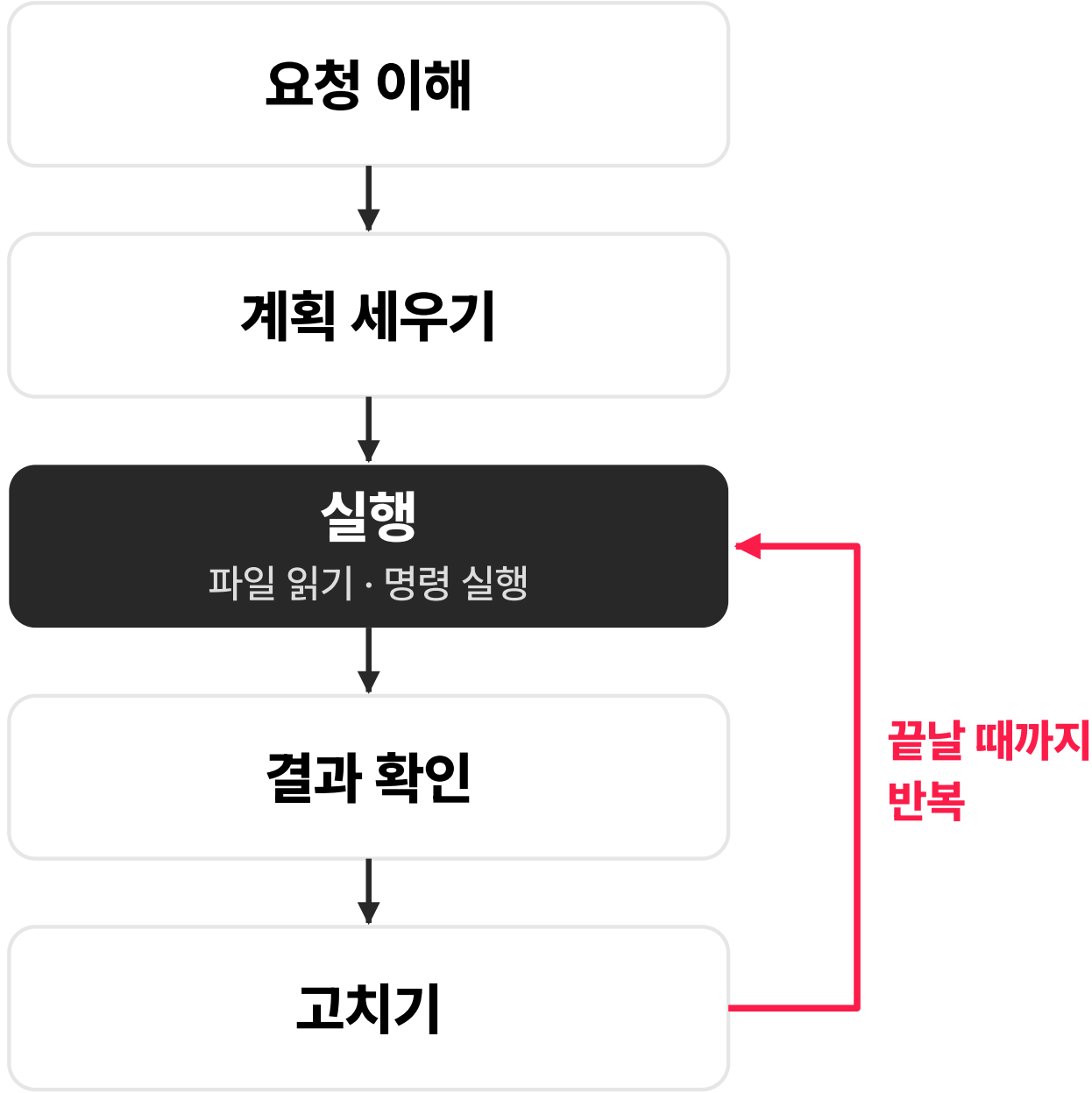
- Bash(curl ... zenodo.org/api/records/6813857)
- Write(proc.py)
- Bash(python proc.py "Waveforms Point05.tdms")
- ✓ point05.csv · time_history.png · spectrum.png

그래프 두 장을 열어 확인했습니다.

파일을 받고, 코드를 실행하고, 결과가 **내 폴더에** 파일로 남습니다.

화면 예시: 왼쪽은 웹 채팅의 일반적인 흐름을 옮겨 그렸고, 오른쪽은 2026-10-13 Claude Code 실제 실행 기록에서 옮겼습니다(실제 캡처로 바꿀 수 있음)

Claude Code는 에이전트 루프로 일합니다



에이전트 루프 · 요청을 받으면 계획하고, 실행하고, 결과를 확인해 고치기를 일이 끝날 때까지 스스로 반복하는 방식입니다.

파일 읽기·검색

파일 만들기·고치기

명령 실행

웹 검색·가져오기

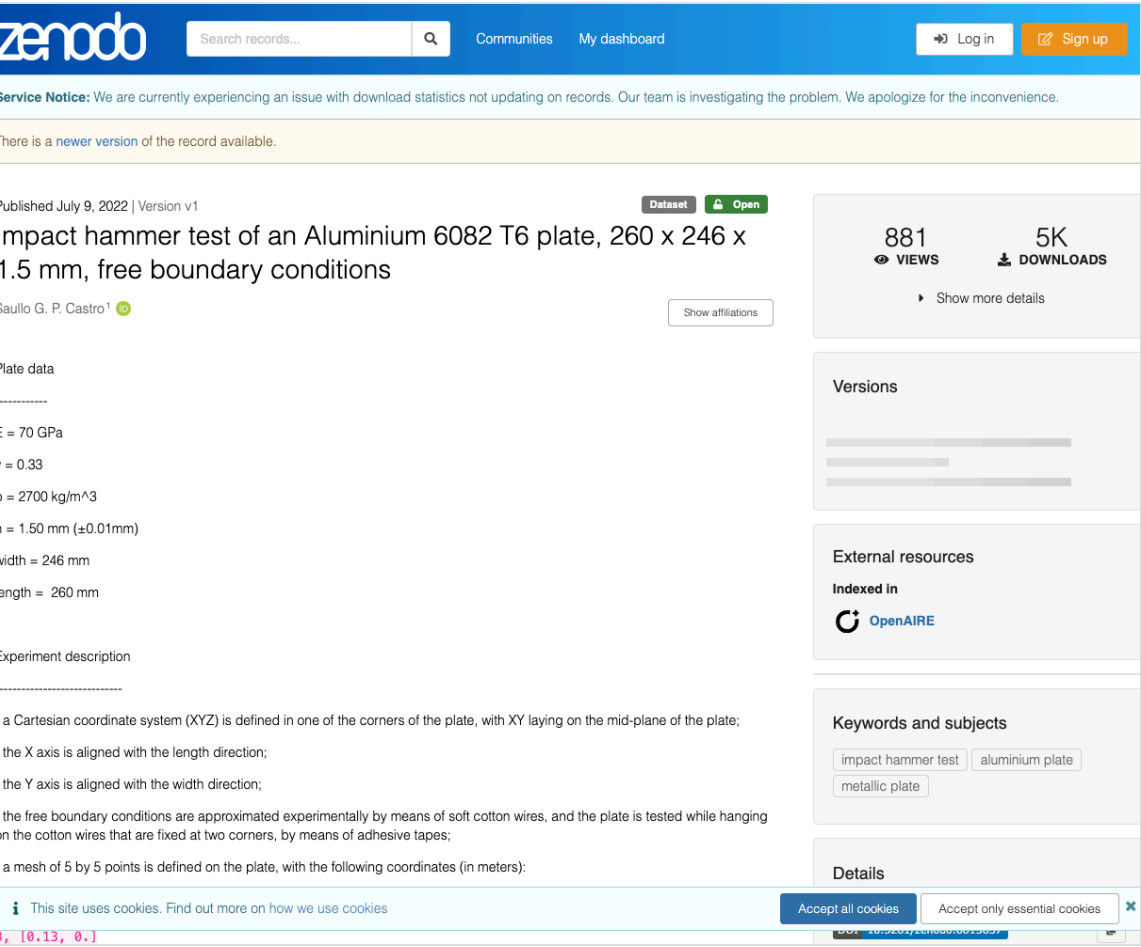
실제 실행 기록 · 고치기가 일어난 곳

```
실행 ● Bash(python proc.py ...)
      └─ ModuleNotFoundError: No module named 'scipy'
확인  설치를 두 번 시도했지만 같은 오류
고치기 "scipy 없이 numpy로 붕우리를 직접 찾습니다."
실행 ● Bash(python proc.py ...)
확인 ● Read(spectrum.png) ✓ 끝
```

1부 · 사전 실행 결과

공개 데이터 주소만 주고 받기부터 그래프까지 맡겼습니다

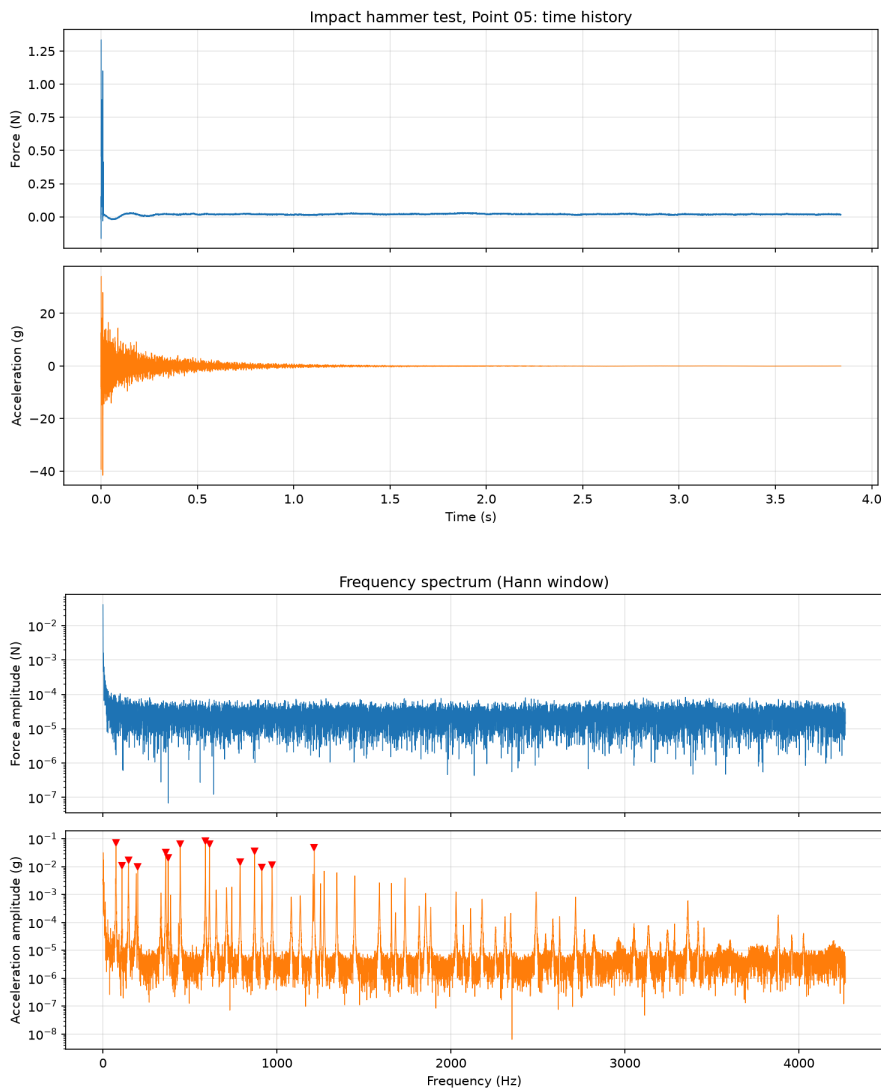
공개 데이터 페이지



알루미늄 판 해머 타격 기록

Zenodo 6813857, Saullo G. P. Castro(2022), CC BY 4.0. 왼쪽은 2026-10-14 원문 캡처, 오른쪽은 2026-10-13 실제 사전 실행 결과

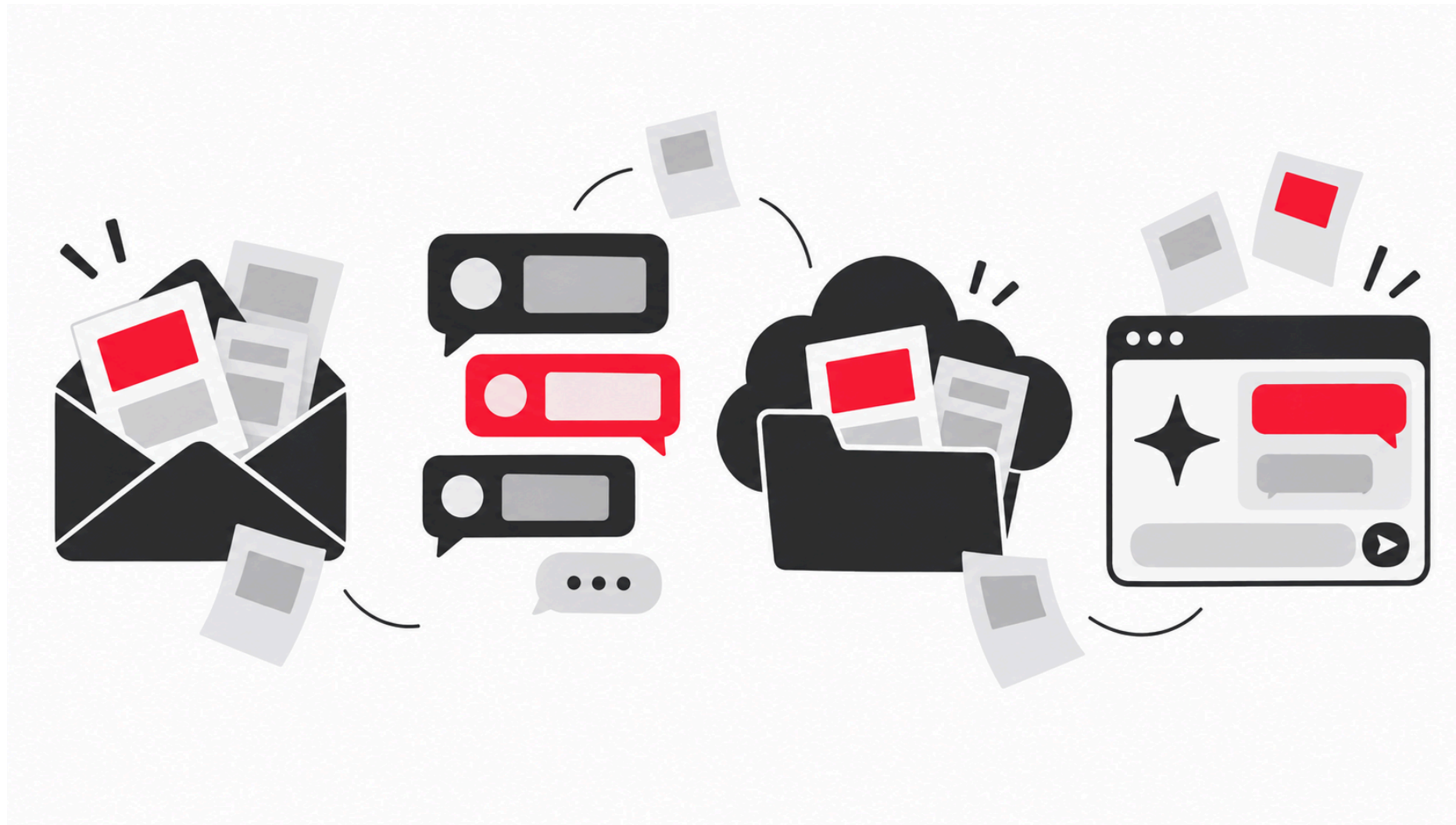
실제 결과



CSV · 시간이력 · 스펙트럼

1부 · 왜 작업 공간이 필요한가

자료와 대화가 채널마다 흩어집니다



메일 · 메신저 · 드라이브 · AI 채팅

같은 업무의 자료가 따로 남습니다.

새 대화에서 다시 설명하고,
자료를 다시 올리고,
지난 답을 다시 찾습니다.

1부 · 내 자료 기준

내가 쌓아 온 자료로 정리하고 분석하고 보고서를 씁니다

내 논문과 보고서

방법·수치·한계를 원문 위치와 함께 정리합니다.

내 측정 파일

형식과 단위를 확인하고 계산·그래프로 바꿉니다.

내 분석 코드

기존 계산을 고치고 다른 방법으로 검산합니다.

내 보고서 양식

근거 있는 결과로 초안을 쓰고, 모르는 칸은 남깁니다.

온라인 조사도 하되, 내 자료·기준·제약을 먼저 읽게 합니다.

1부 · 내가 쓰는 작업 공간

연구 자료와 일하는 순서를 한 폴더에 담았습니다

research-planner

강사가 만든 연구 기획 워크스페이스입니다.
오후에는 각자 PC에서 내 것으로 만듭니다.

/onboarding 인터뷰에 답하고,
내 자료는 **00-inbox/**에 넣습니다.

규칙 · CLAUDE.md

판단 기준 · frameworks/

작업 순서 · .claude/commands/

계산·검사 · 60-scripts/

자료와 기록 · 70-kb/ · 90-log/

1부 · 축적의 예

처음부터 완벽하지 않고, 쓸수록 맞아집니다



1주 · 시작

내 자료와 양식
잘못 이해한 점



1개월 · 반복

자주 쓰는 요청
고친 코드와 기준

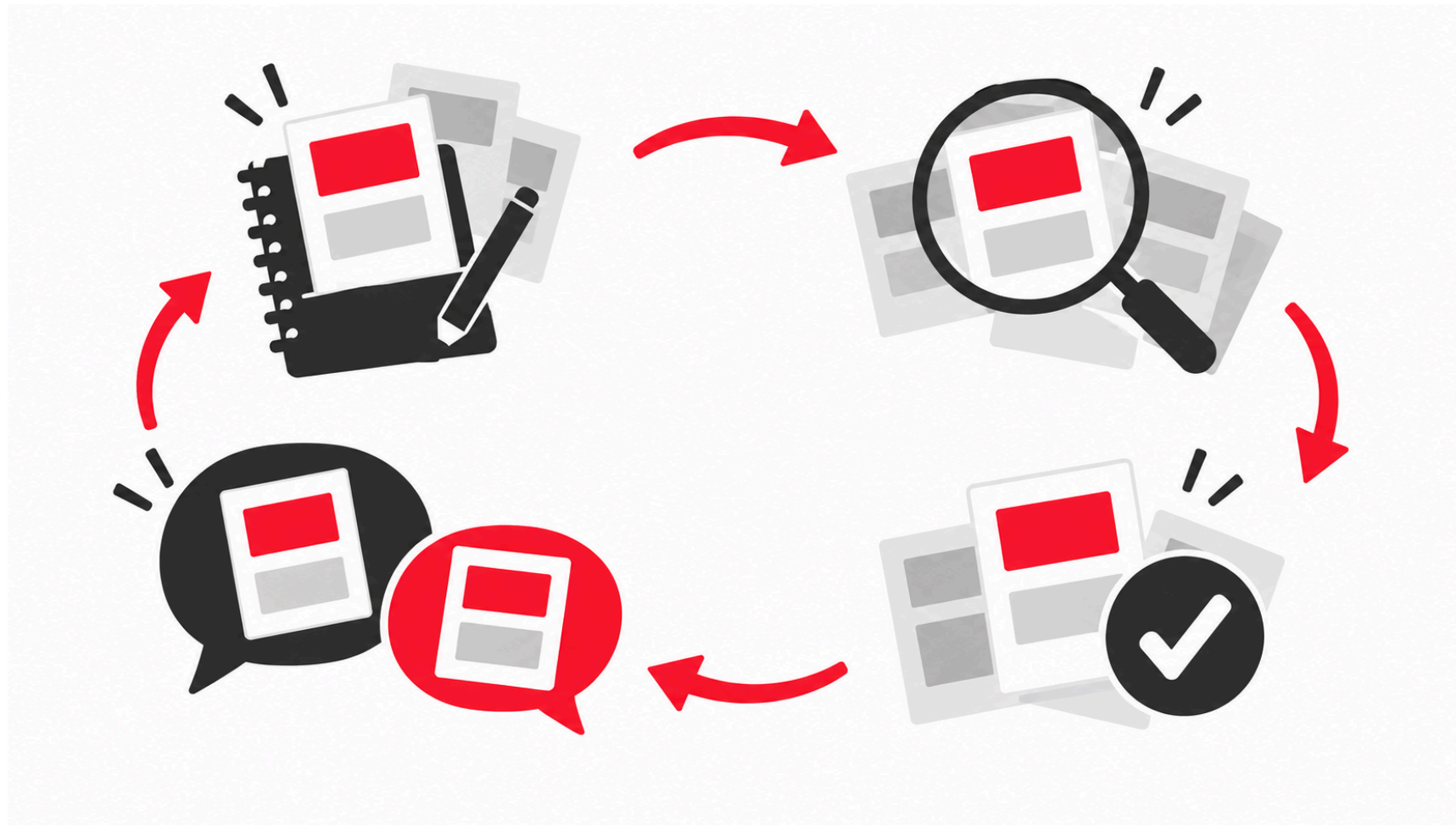


3개월 · 축적

검증한 결과
다음 작업의 출발점

1부 · 맥락을 관리하는 환경

기록을 다음 작업에서 다시 읽게 만듭니다



1. 기록한다

무엇을 했고 무엇이 남았는지 저장

2. 찾는다

이번 작업에 필요한 자료만 검색

3. 확정한다

검증한 사실과 미확인을 구분

4. 다음 세션에서 읽는다

지난 결정과 다음 할 일부터 시작

1부 · 안전하게 쓰기

처음에는 모든 변경을 직접 승인하는 모드로 씁니다

Manual

파일 수정과 명령 실행마다
문습니다

오늘 실습의 기본

Accept edits

파일 수정은 바로 하고, 명령
실행은 문습니다

익숙해진 뒤

Auto 계열

대부분 묻지 않고 진행합니다

오늘은 쓰지 않음

밖으로 나가는 자료

요청문과 읽은 파일은 Anthropic 서버로 갑니다. 기관
자료는 승인한 환경에서만.

되돌리기의 한계

명령 실행으로 바뀐 파일은 되돌리지 못합니다. 작업
전에 폴더를 복사해 둡니다.

2부

연구개발 활용 사례와 플러그인 조합

연구 업무를 에이전트에 맡긴 실제 결과를 보고, 바로 쓸 수 있는 조합을 소개합니다

연구자는 반복하던 분석·계측 프로그램부터 만들고 있습니다

RIKEN · 현미경 프로그램

LabVIEW 앱을 Python으로 다시 작성

카메라·광원·스테이지를 연결하고
이동 한계는 사람이 정했습니다.

NREL · 소스미터 프로그램

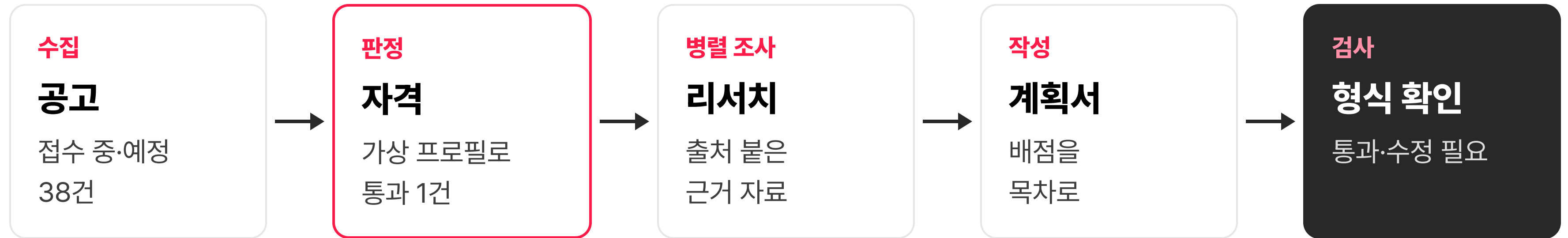
제어 코드와 화면을 AI로 작성

실험 설계와 결과 검증은
연구자가 맡았습니다.

반복되는 작업을 프로그램으로 만들되, 실험 조건과 검증은 연구자가 맡습니다.

2부 · KBSI 세미나에서 보여 준 흐름

공고를 모으고, 자격을 확인한 뒤 계획서를 씁니다



형식 통과 ≠ 사실 확인

형식 검사를 통과한 문서에서도 법령 오독과 비용 계산의 모순이 나왔습니다.

검사 코드는 형식을 확인하고, 사람은 원문과 사실을 확인합니다.

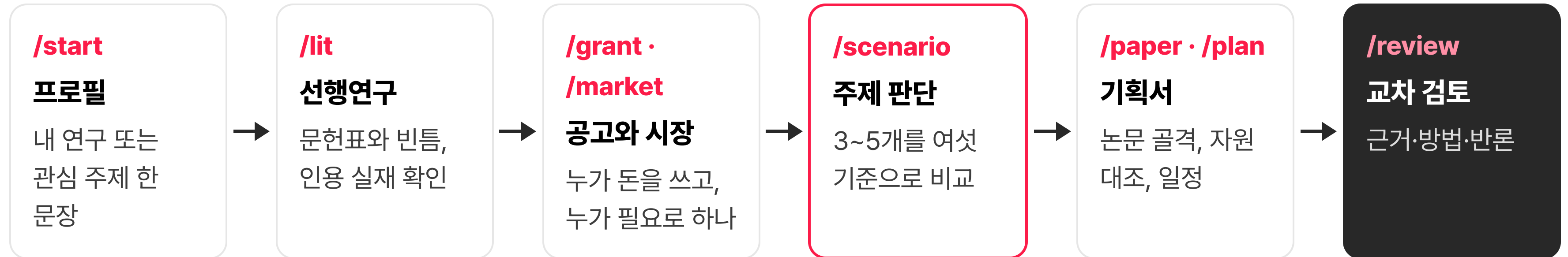
오후에는 이 조합을 직접 써 봅니다

하는 일	플러그인·명령	남는 것
자료 읽기·조사	insane-search · insane-research · /lit	출처와 문헌표
공고 찾기·적합도	/grant search · /grant fit	공고 목록과 판정표
분석 코드 고치기	docs-guide	공식 문서와 그래프
도구 기획하기	show-me-the-prd	기획 문서
반복 업무 묶기	skillers-suda	내 스킬
결과 교차 검토	kkirikkiri	검토 의견과 수정

오후에는 내 자료로 워크스페이스를 만들고, 이 조합으로 논문부터 공고 적합도·기획서까지 해 봅니다.

2부 · 연구 기획 워크스페이스

조합을 한 폴더에 담아, 하고 있는 연구에서 다음 연구를 기획합니다



폴더를 열면 플러그인이 같이 켜집니다

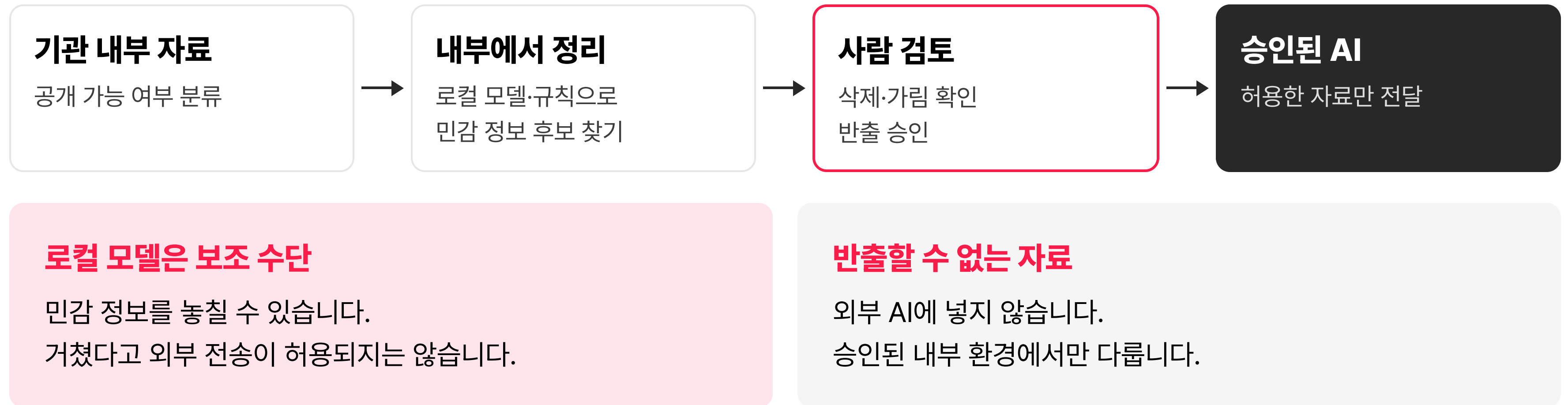
폴더 실행 창에서 허용하면 insane-research, insane-search, docs-guide, show-me-the-prd, kkirikkiri, skillers-suda 가 자동으로 등록됩니다. GitHub에 접속할 수 없으면 플러그인 없이도 명령이 동작합니다.

과제 공고는 기능 하나로

앞에서 본 research-lab의 공고 검색, 적합도 판정, 계획서 검사를 **/grant**, **/proposal**로 옮겼습니다. 하지 않은 실험의 수치는 쓰지 않고, 증빙 없는 "보유"는 검사 스크립트가 잡습니다.

2부 · 자료 보안

자료를 넣기 전에 기관의 사용 지침부터 확인합니다



3부

장비 연동: 컨트롤러와 프로토콜

제조사가 공개한 통신 방법으로 내 업무에 맞는 프로그램을 만들 수 있습니다

3부 · 전과 후

제조사 프로그램을 쓰는 데서, 내 작업용 프로그램을 만드는 쪽으로



전 · 주어진 기능에 맞춰 일하기

제조사 프로그램으로 조작하고
없는 기능은 기다리거나 손으로 처리

후 · 필요한 기능을 직접 만들기

공개 매뉴얼·SDK를 AI와 읽고
내 순서·기록·보고서에 맞게 작성

AI가 직접 다룰 때는 MCP 같은 도구 통로로 연결하고, 허용 명령과 범위는 별도로 제한합니다.

시험·계측 장비는 대개 네 가지 방식 중 하나로 PC와 통신합니다

방식	연결	주로 쓰는 장비	Python에서 부르는 법
RS-232(시리얼)	COM 포트와 케이블, 통신 속도·패리티 설정	구형 계측기, 제어기, 센서 변환기	pyserial, PyVISA(ASRL 주소)
USB	USB-TMC 또는 가상 COM 포트	오실로스코프, 멀티미터, 전원 장치	PyVISA(USB 주소)
TCP/IP(LAN)	IP 주소와 포트. LAN 계측기는 HiSLIP(4880), VXI-11, 원시 소켓(관례상 5025) 중 하나	LAN 계측기, DAQ, 계측 제어기	PyVISA(TCPIP 주소), socket
제조사 API·SDK	REST/HTTP, Windows COM, 자체 TCP 프로토콜, gRPC	계측 제어기, DAQ 소프트웨어	제조사 Python 패키지와 예제

계측 제어기의 원격 제어는 유료 옵션이거나 명령 명세가 공개되지 않은 경우가 많습니다. 먼저 내 장비가 어떤 인터페이스를 제공하는지, 라이선스가 있는지 확인합니다.

AI 이전에도 연구실은 공개 API로 수집·시험을 자동화했습니다

류블라나대 · LDAQ

DAQ·Arduino·열화상 카메라 등
서로 다른 장비의 데이터 수집을
Python 패키지 하나로 묶었습니다.

브리스톨대 · Instron 시험

Python에서 기존 제어 프로그램을 거쳐
반복 시험과 측정 기록을 자동화했습니다.
실행 전 한계값은 따로 확인합니다.

달라진 점은 연결 원리가 아니라, 이 프로그램을 AI와 함께 만들 수 있다는 것입니다.

이제는 Claude Code 같은 에이전트가 그 프로그램을 같이 만듭니다

누가	무엇을 만들었나	사람이 정한 것
RIKEN 호시나 연구원 (2026-02)	Claude Code로 LabVIEW 현미경 앱(카메라·광원·스테이지)을 Python으로 다시 만듦. 카메라 모듈 약 5분, 전체 몇 시간(본인 보고)	자동초점 1회 이동 최대 1mm
정밀 계측 개인 사용자 (2026-05)	Claude Code로 GPIB 계측기 6대의 제어·기록·교정 프로그램을 만듦. 사양 문서는 사람이 먼저 씀	최대 1kV 설정에 안전 게이트와 운용자 확인
미국 NREL 재료과학센터	ChatGPT-4로 Keithley 2400 소스미터 제어 코드와 GUI를 만들어 공개 (Digital Discovery 2025)	실험 설계와 결과 검증
중국 베이항대 (2026)	시험 의도를 말로 받아 여러 계측기의 SCPI 코드를 만드는 다중 에이전트. 실행 성공률 82.3%(저자 보고)	실행 전 사람 검토 단계

주의 사례: 원자간력현미경을 조작한 LLM 에이전트가 지시하지 않은 위험한 탐침 이동을 했고, 저자들은 이를 "몽유(sleepwalking)"라고 보고했습니다(Nature Communications 2025).

제조사도 AI가 계측기를 다루는 연결 도구를 공개하고 있습니다

계측기 제조사

Keysight·Tektronix의 공개 사례는
AI 요청을 계측기 명령으로 연결합니다.

▶ [Tektronix 공식 영상 열기](#)

OPC UA 연결

OPC Foundation이 산업용 통신 규약
OPC UA와 AI 도구를 잇는
MCP 서버를 공개했습니다.

도구 연결과 안전 운용은 별개입니다. 읽기 전용부터 시작합니다.

먼저 현재 상태를 읽습니다. 시뮬레이터는 필수가 아닙니다

사진·화면·소리에서 읽기

멀티모달: 여러 종류의 입력을 함께 해석

표시창 숫자, 계기 바늘, 경고등을 읽어
현재 상태를 빠르게 정리합니다.

센서·API에서 값 받기

센서 측정값과 장비 응답을 직접 수집

온도·전류·운전 모드와 시각을 묶어
판단에 쓸 상태를 만듭니다.

둘을 함께 쓸 수 있습니다. 값의 단위·측정 시각·누락 여부를 먼저 확인합니다.

3부 · 판단과 시뮬레이션

“바꾸면 어떻게 될까”를 예측하려면 모델이 더 필요합니다



상태 조화·알림부터

기록·점검·이상 후보 찾기는
완전한 시뮬레이터 없이 시작할 수 있습니다.

예측을 조작 근거로 쓸 때

실측과 대조해 모델 오차를 확인합니다.
예측 성공이 실제 안전을 보장하지는 않습니다.

장비를 에이전트에 연결할 때는 네 가지를 기본 요건으로 둡니다

1 동작 전 사람 승인	2 동작 상한값 제한	3 비상 정지와 중단	4 명령과 결과 기록
장비를 움직이는 명령은 사람이 확인한 뒤에만 보냅니다. 어디서 지키나 Claude Code 도구 권한 확인 창, 드라이버의 승인 함수	자유도별 움직임, 출력 레벨, 주파수 범위에 상한을 둡니다. 어디서 지키나 장비 제어기(1차), 드라이버 코드(2차, 더 보수적으로)	이상이 감지되면 즉시 멈춥니다. 정지는 승인 없이 부를 수 있습니다. 어디서 지키나 장비 비상 정지 회로, 드라이버 ·MCP의 정지 도구	누가, 언제, 무엇을 보냈고 어떤 응답을 받았는지 남깁니다. 어디서 지키나 드라이버 명령 로그, Claude Code 혹은 실행 이력

상한과 정지는 에이전트가 고칠 수 없는 드라이버와 제어기에 두고, 승인과 기록을 그 위에 더합니다. 모의 장비 실습에서 승인·상한·정지·기록을 확인합니다.

근거: 과업지시서 5.다.2) 안전 운용 원칙. 기관 장비의 운용 안전수칙을 대신하지 않습니다

4부

피지컬 AI와 Gemini Robotics

로봇에서는 AI가 어디에 쓰이고, 안전은 어디에 두는지 봅니다

업계는 느린 판단과 빠른 제어를 나누고, 안전은 여러 겹에 둡니다

상위 추론

느림

상황을 이해하고 다음에 할 일을 고릅니다. 대화형 모델, Jev 같은 판단 모델이 여기에 있습니다.

하위 행동

빠름

고른 일을 실제 움직임으로 바꿉니다. 손으로 짤 코드인 경우도 있지만, 작은 학습 모델인 경우가 많습니다.

최하위 제어·안전

가장 빠름

모터 제어, 인터락, 비상 정지가 있습니다. 상위 계층이 실수해도 여기서 멈춥니다.

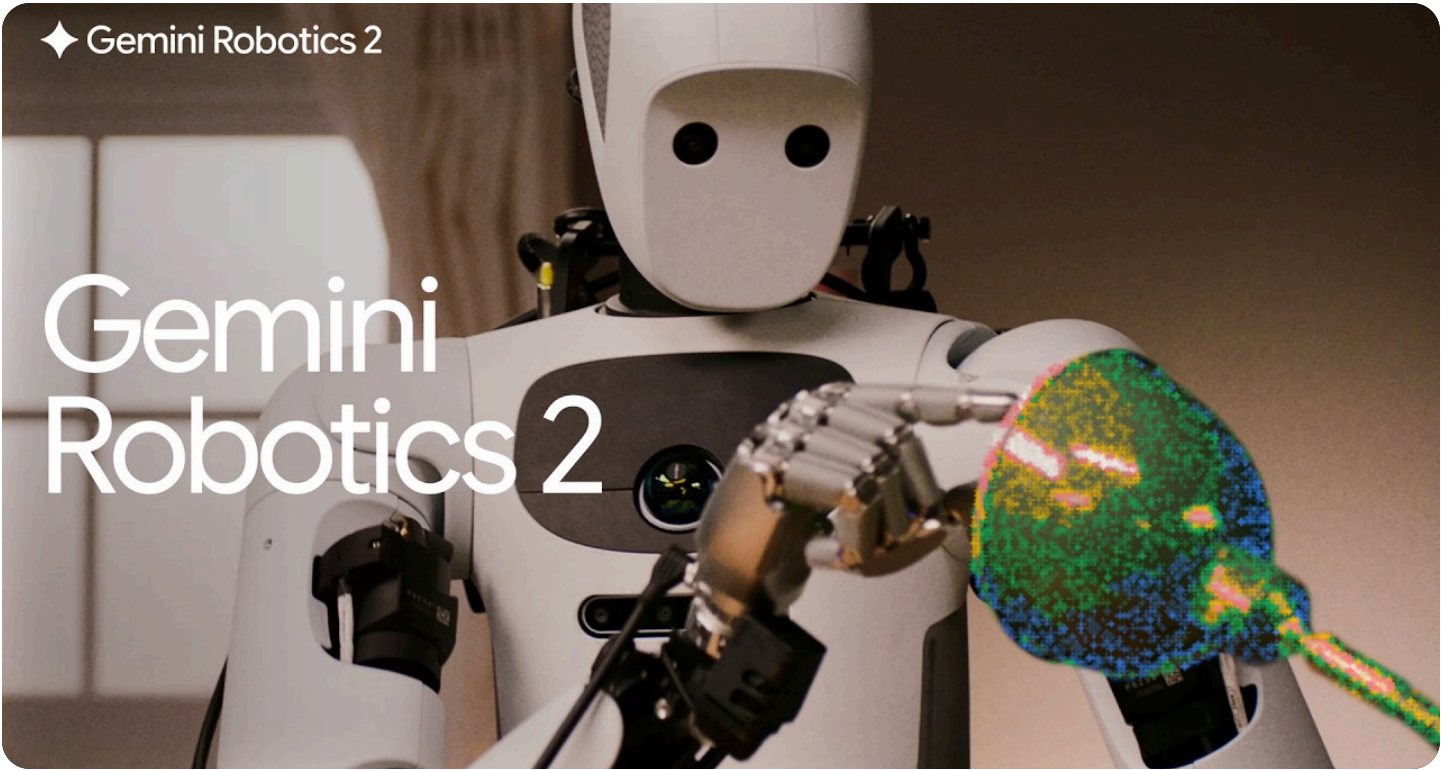
공식 자료에서 같은 구조를 설명하는 곳

- Figure Helix
- NVIDIA GR00T
- Google DeepMind Gemini Robotics
- Physical Intelligence $\pi 0.6$

이들 자료는 안전 장치를 최하위 제어 계층과 상위 모델의 안전 정지 호출 양쪽에 둔다고 설명합니다.

1년 반 사이 종이접기에서 휴머노이드의 방 정리까지 왔습니다

● 2025-03	● 2025-06	● 2025-09	● 2026-07
Gemini Robotics	On-Device	1.5	2
종이접기, 지퍼백에 간식 넣기, 지시대로 책상 정리	네트워크 없이 로봇 안에서 실행. 산업용 벨트 조립	빨래 색깔별 분류. 한 로봇에서 배운 동작을 다른 로봇에 이전	휴머노이드 전신 동작으로 방 정리, 여러 로봇이 일을 나눠 완성



- 회사도 "여러 손가락으로 하는 정밀 작업은 아직 어렵고 동작 속도도 개선이 필요하다"고 적었습니다.
- 실제 시연 데이터가 비싸서 시뮬레이션과 사람 영상으로 메우고, 시뮬레이션과 현실의 차이는 회사들도 한계로 적습니다.

4부 · 도구 목록

AI는 주어진 도구 목록 안에서만 로봇을 움직입니다



▶ 이미지를 누르면 영상 · Spot과 Gemini로 할 일 목록 처리하기 (Boston Dynamics, 1분 50초, 실험 단계)

Gemini가 부를 수 있던 도구 네 개

GoTo	이동
TakePicture	사진 찍기
pickup	집기
PutDown	내려놓기

- 걷기, 길찾기, 잡기 동작은 Spot의 기존 소프트웨어가 맡았습니다.
- Boston Dynamics는 Gemini가 "API에 있는 것 이상으로 Spot을 제어할 수 없다"고 적었습니다.

시험 장비로 옮기면 · 도구 목록은 드라이버가 허용한 명령이고, 상한과 정지는 제어가 지킵니다. 3부의 장비 연결 구조와 같습니다.

출처: bostondynamics.com/blog/tools-for-your-to-do-list-with-spot-and-gemini-robotics. 영상 youtube.com/watch?v=LP4-c5AK30g. Boston Dynamics가 "아직 실험 단계"라고 밝힘

5부

Jev

모델은 선택지에서 고르고, 빠른 제어와 안전은 아래 계층이 맡습니다

Jev는 글을 쓰지 않고 정해진 선택지 중 하나를 고릅니다

대화형 모델(Claude·GPT 등)

- 문장을 써서 답합니다
- 중간 과정을 글로 따져 볼 수 있습니다
- 선택지 하나를 고르는 데도 몇 초가 걸립니다

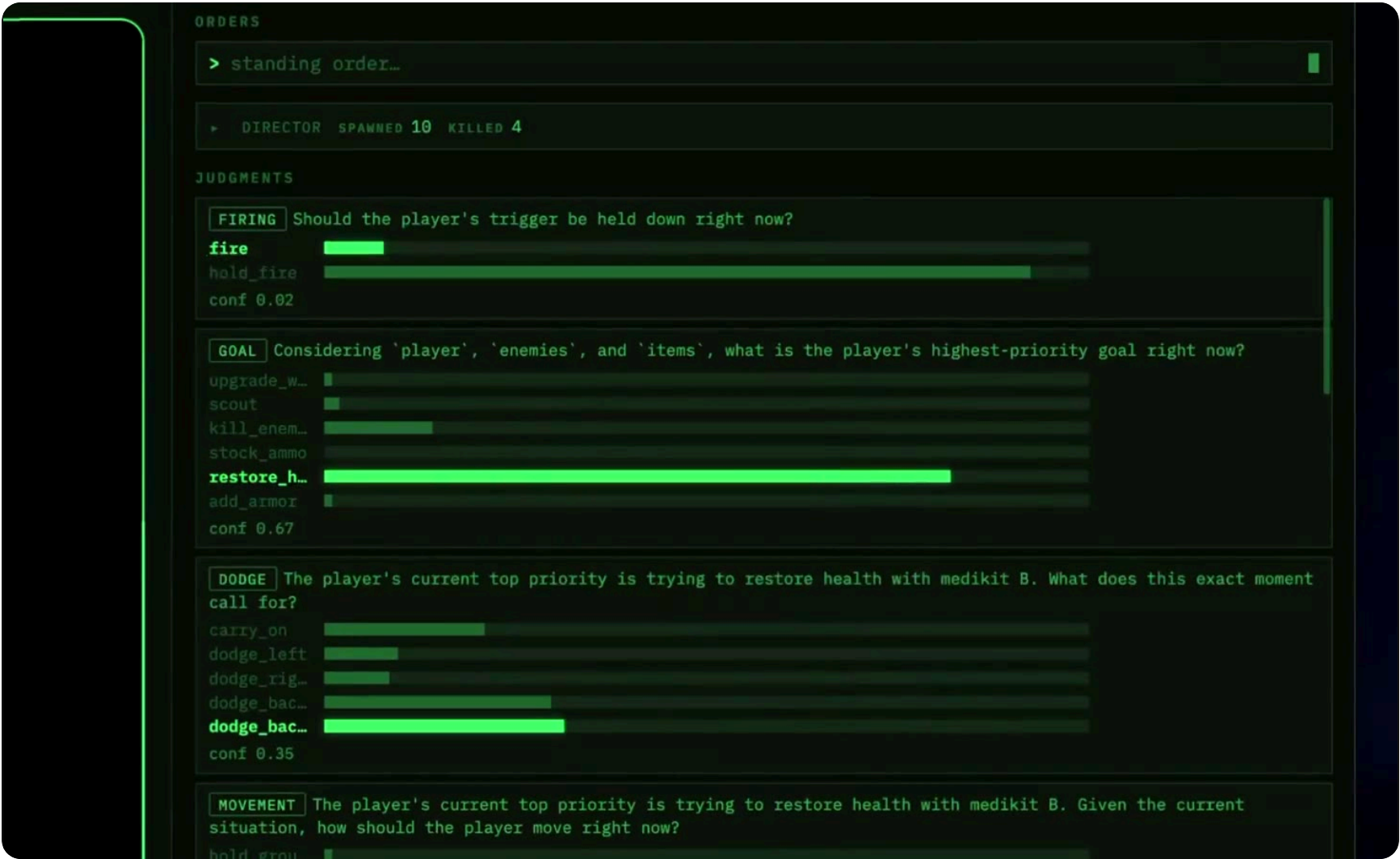
Jev(TypeSafe, 2026-09-15 공개)

- 코드가 정한 선택지 중 하나와 그 확률을 돌려줍니다
- 응답에 약 70~500ms가 걸립니다
- 회사 스스로 숫자 세기, 산술, 날짜 비교에 약하다고 밝혔습니다

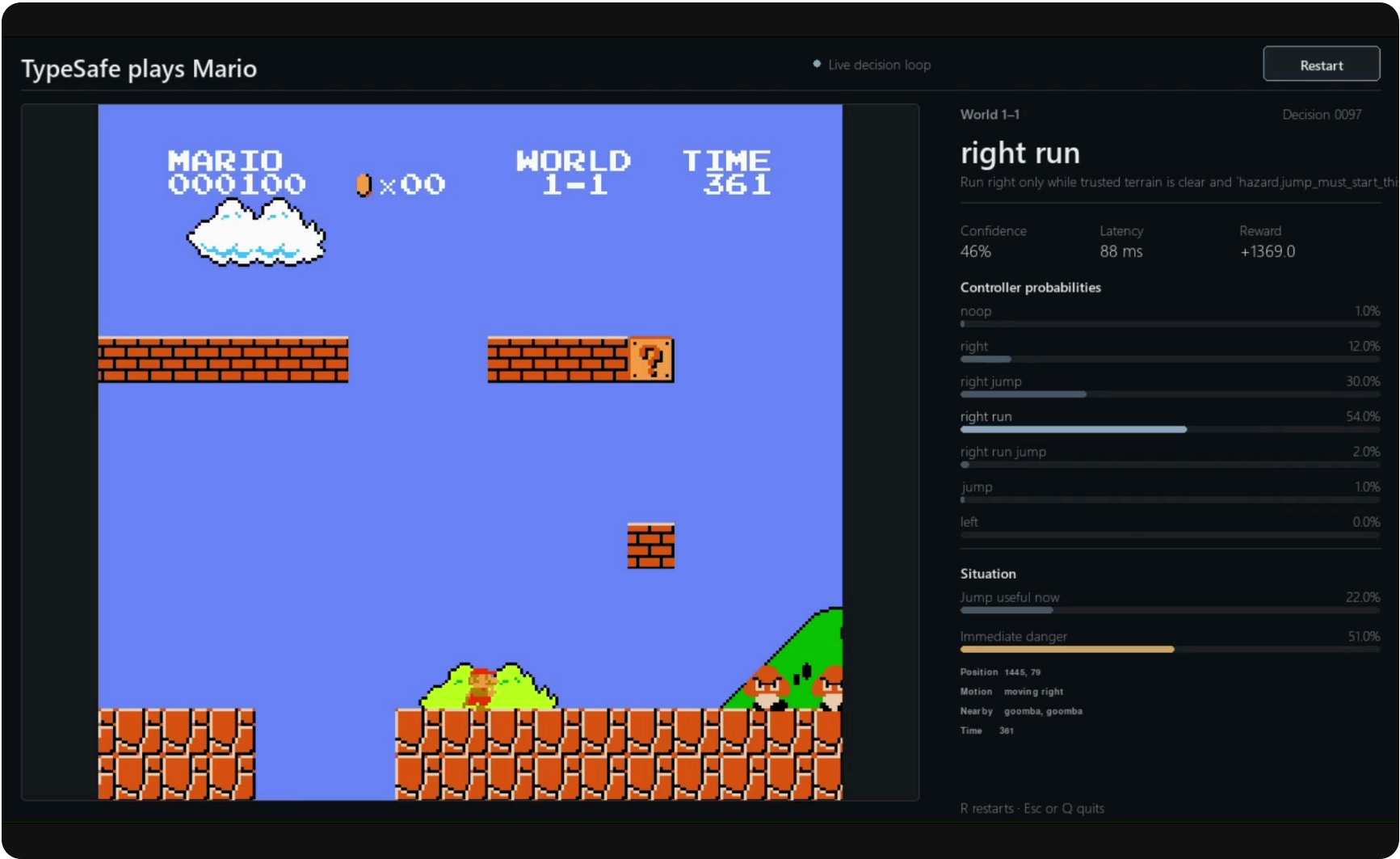
그래서 Jev는 제어기 위에서 빠르게 판단하는 자리에 놓입니다. 다음 영상들도 같은 구조입니다.

5부 · 영상 1 · 2

게임 상태를 글로 넘기면 Jev가 다음 조작을 고릅니다



▶ Doom · 클릭하면 원본 영상 · 상태를 글로 전달 · 공식 데모



▶ 슈퍼마리오 · 클릭하면 원본 영상 · 버튼 조합 선택 · 커뮤니티 데모

오전에 본 것을 네 줄로 정리합니다

1. Claude Code

내 폴더의 파일을 읽고, 코드를 실행하고, 결과를 파일로 남기는 에이전트입니다. 규칙과 절차를 폴더에 담으면 워크스페이스가 됩니다.

2. 연구 업무

계산은 코드, 서술은 AI, 판정은 사람으로 나눠 맡기고, 다른 방법으로 검산합니다.

3. 플러그인 조합

조사, 코드, 도구 만들기, 스킴, 교차 검토를 플러그인 조합으로 맡기고 결과를 파일로 남깁니다.

4. 장비

제조사가 공개한 통신 방법으로 직접 프로그램을 만들 수 있습니다. 상한과 정지는 에이전트가 바꿀 수 없는 드라이버와 제어기에 둡니다.

오후 안내

오후에는 내 PC에 연구 워크스페이스를 직접 세팅합니다

1. 빈 폴더에서 처음 맡겨 보기
2. 인터뷰와 내 자료로 연구 워크스페이스 만들기
3. 주제 하나로 논문·공고 적합도·기획서까지
4. 스킬과 기록 자동화로 계속 쓰기

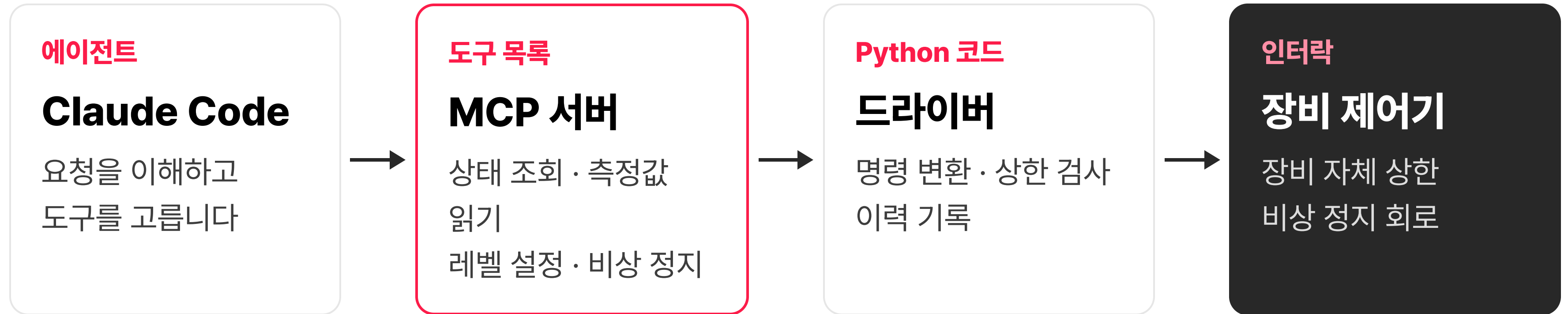
준비 · 실습 안내 kimm.axwith.com, 앱 로그인, 써 본 분은 내 공개 자료 10개 이내

부록

시간이 남으면 보는 장

더 깊은 사례와 장비 연동 절차, 영상을 모았습니다

MCP는 에이전트가 쓸 수 있는 도구 목록을 장비 쪽에서 정해 줍니다



- 이 MCP 연결에서는 서버가 공개한 도구를 부릅니다. 별도 터미널·네트워크 접근 권한도 함께 제한해야 합니다.
- 도구마다 사람의 승인을 거치게 할 수 있습니다. 읽기는 바로, 장비를 움직이는 명령은 매번 확인 창이 뜨게 설정합니다.
- MCP 대신 명령줄 도구(CLI)로 만들어도 됩니다. 에이전트가 터미널에서 그 도구를 실행합니다.

가상 장비로 네 요건을 차례로 보여 드립니다

순서	해 보는 것	화면에서 확인할 것
0	가상 계측기와 출력 장치 연결, 상태 읽기	읽기 명령은 승인 없이 바로 실행됩니다
1. 사람 승인	출력 레벨 설정을 요청하고, 확인 창에서 거절	거절하면 명령이 장비로 가지 않습니다
2. 상한값	상한을 넘는 레벨과 축 변위를 요청	드라이버가 보내기 전에 거부하고, 장비도 범위 밖 값을 거부합니다
3. 비상 정지	측정 피크가 기준을 넘는 상황을 가정하고 정지 호출	출력이 꺼지고 레벨이 0이 됩니다. 정지는 승인 없이 실행됩니다
4. 이력 기록	명령 로그와 혹 이력 파일 열기	보낸 명령, 응답, 거부, 승인 여부가 시각과 함께 남아 있습니다

실장비가 아니라 pyvisa-sim으로 만든 가상 장비입니다. 오후 더 해 보기에서 같은 가상 장비를 연결합니다.

2부 · 역할 나누기

측정 후 작업은 계산, 서술, 판정을 나눠 맡깁니다

코드가 할 일

- 원시 데이터 파싱
- 스펙트럼·전달함수 계산
- 합격·불합격 규칙 적용

AI가 할 일

- 시험계획서·기준서에서 요구사항 뽑기
- 요구사항과 측정 증거 대조표
- 보고서 서술 초안, 빠진 증거 찾기

사람이 할 일

- 모든 판정
- 보고서 검토와 서명
- 기준 해석과 예외 판단

research-lab의 연구 규칙을 시험 보고서 규칙으로 바꾸면

- 측정하지 않은 값은 추정해 채우지 않고 "미측정"으로 남깁니다.
- 보고서의 모든 수치는 실제로 실행한 계산 출력과 일치해야 합니다. 문장 속 숫자를 손으로 고치지 않습니다.

2부 · 조합 시연

공식 문서 확인과 문헌 조사를 차례로 보여 드립니다

시연 1 · docs-guide (라이브)

npTDMS 공식 문서 기준으로
채널의 단위와 샘플 간격 속성을
읽는 코드를 알려 줘.

- 공식 문서 주소를 먼저 가져오고, 그 내용에 근거해 답합니다
- 1부 시연에서 만든 스크립트와 비교합니다

시연 2 · insane-research (미리 돌린 결과)

측정 데이터 후처리와
보고서 자동화의 공개 사례를
출처와 함께 조사해 줘.

- 여러 에이전트가 찾고, 출처를 서로 대조한 보고서가 남습니다
- 보고서의 인용은 원문 링크를 열어 사람이 확인합니다

웹을 쓰는 플러그인에는 공개 주제만 넣습니다. 기관 과제명·장비 정보·시험 데이터를 검색어나 요청문에 넣지 않습니다. pumasi-insane-review는 자료를 Anthropic이 아닌 다른 회사로도 보내므로 기관 자료에는 쓰지 않습니다.

2부 · 로컬 모델

같은 에이전트 도구에 모델만 로컬로 바꿔 끼울 수 있습니다

실제로 돌려 봤습니다 (2026-10-01, 맥)

- Ollama는 Anthropic Messages API 호환 주소를 제공해, 주소만 바꾸면 같은 에이전트 도구가 로컬 모델로 돕니다. 도구 호출도 동작했습니다.
- 35B 로컬 모델로 파일 만들기 작업이 실제로 됐습니다. 다만 첫 응답까지 약 104초가 걸렸습니다.

온프레미스 후보 모델

- Qwen3-Coder-30B-A3B(단일 24~48GB GPU), gpt-oss-20b·120b가 기본 후보입니다.
- vLLM이 이 모델들의 도구 호출을 지원합니다.
- 같은 조건의 비교 평가는 없어 "후보"로만 소개합니다.

도입 단계 제안

1. 개인 PC

Ollama와 작은 모델, 공개 데이터로 익히기

2. 부서 공용

사내 GPU 서버에 vLLM, 사내망 전용, 인증과 로그

3. 서비스

판단이 필요한 곳과 제어가 필요한 곳을 나눠 구성

3부 · 도구로 만드는 절차

장비 기능을 에이전트 도구로 만드는 일반 절차는 다섯 단계입니다

- 1. 매뉴얼과 연결 정보 확보**

장비 프로그래밍 매뉴얼, 연결 방식(IP 주소, 포트, VISA 주소)을 준비합니다.
- 2. 필요한 명령만 골라 드라이버 작성**

에이전트에게 드라이버를 짜게 하고, 명령마다 매뉴얼 쪽수를 주석으로 달게 합니다.
- 3. 모의 장비와 읽기 전용으로 시험**

가상 장비나 조회 명령으로만 먼저 돌려 봅니다.
- 4. 상한, 승인, 이력 추가**

동작 명령에 코드 상한을 걸고, 사람 승인과 명령 기록을 넣습니다.
- 5. MCP 서버나 CLI 도구로 등록**

검증한 드라이버를 도구로 감싸 에이전트가 다시 쓰게 합니다.

경험자 한 줄 · 제조사 공식 예제가 오래됐다면, 매뉴얼을 주고 현행 라이브러리로 다시 짜게 하는 것도 같은 절차입니다. 2부의 docs-guide로 라이브러리 공식 문서를 함께 확인하게 합니다.

3부 · 나눠 드리는 체크리스트

내 장비의 인터페이스를 여덟 가지로 확인해 보세요

1. 제어 소프트웨어가 원격 제어를 지원하나요? (REST, COM, 자체 TCP, SCPI/VISA, 모름)
2. 자동화 기능이 라이선스 옵션인가요? 보유하고 있나요?
3. 명령 명세나 매뉴얼을 갖고 있나요? (파일명·쪽수)
4. 시뮬레이션·데모 모드가 있나요?
5. 장비가 내보내는 데이터 파일 형식은? (TDMS, UFF, HDF5, MDF, CSV 등)
6. 레벨·주파수 상한과 비상 정지는 어디에 설정되어 있나요?
7. 장비 PC가 사내망에 연결돼 있나요? 외부망은요?
8. 먼저 자동화하고 싶은 일 하나 (측정 전, 측정 중, 측정 후)

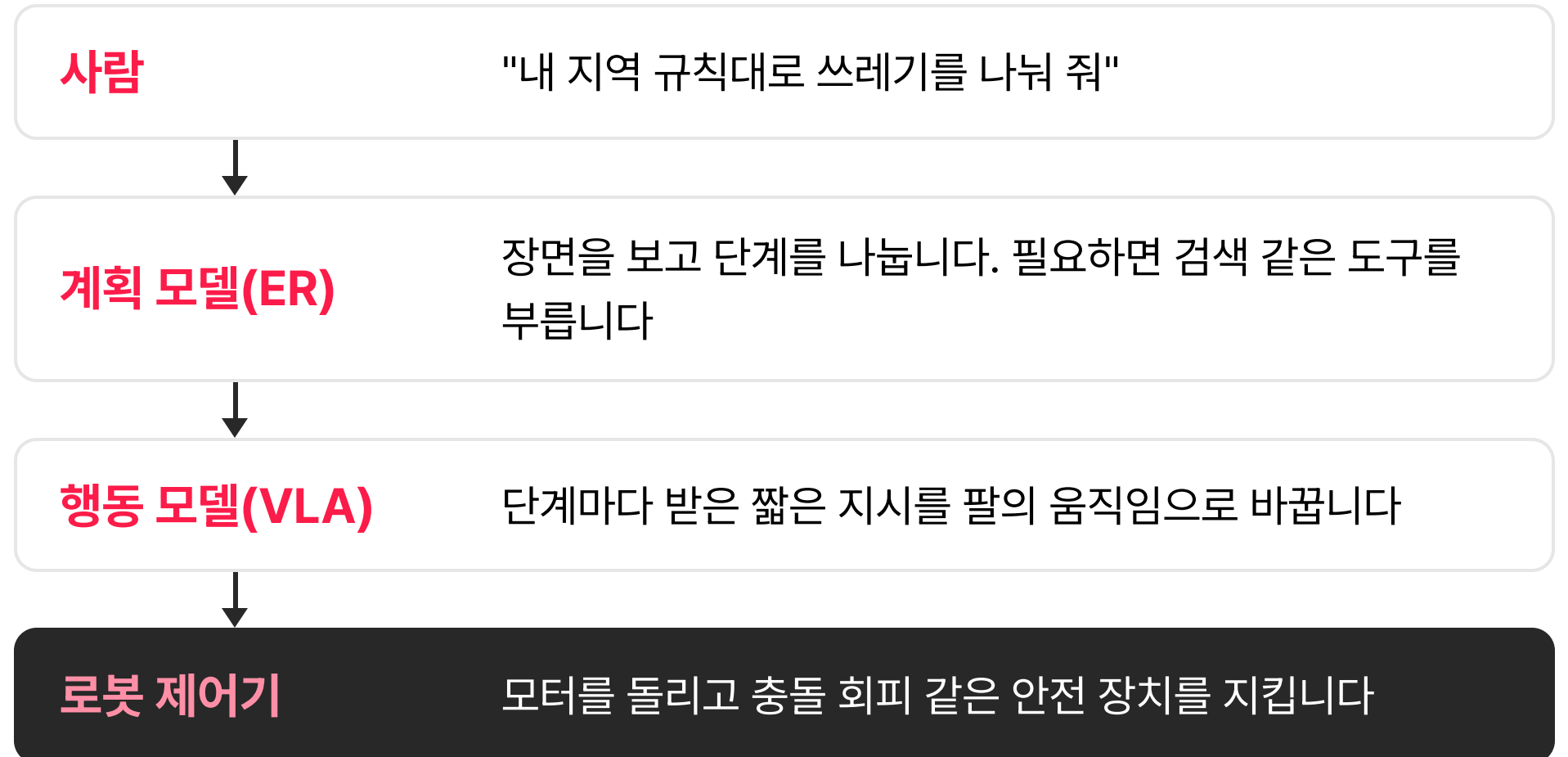
이 체크리스트는 가져가서 씁니다 · 오후 실습은 장비 정보 없이 공개 주제로 합니다. 1~4번이 채워진 분은 오후 "더 해 보기"에서 모의 장비 드라이버에 읽기 명령을 하나 더해 보시고, 기관에 돌아가 내 장비로 같은 절차를 밟을 때 이 표를 출발점으로 씁니다.

4부 · Gemini Robotics 구조

Gemini Robotics는 계획하는 모델과 움직이는 모델을 나눠 둡니다



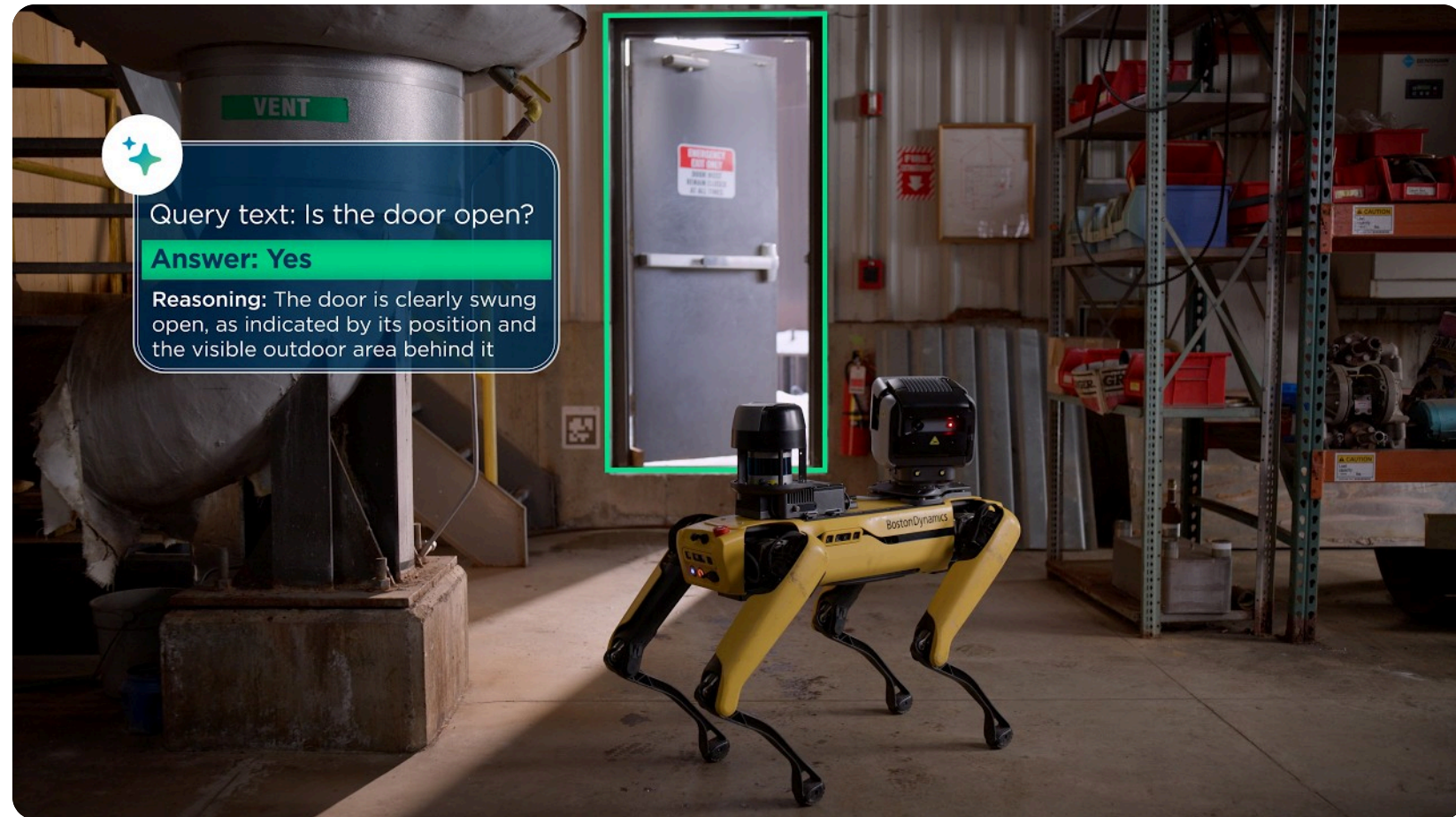
▶ 이미지를 누르면 영상 · 지역 분리배출 규칙을 검색한 뒤 쓰레기를 나눕니다
(Google DeepMind, 1분 42초)



일반 개발자에게 API로 공개된 것은 계획 모델(ER)뿐입니다. 공식 문서에는 "로봇 주변의 안전한 환경 유지는 사용자 책임"이라고 적혀 있습니다.

4부 · 현장 적용 · 계기 읽기

현장에 먼저 들어간 것은 계기를 읽는 일입니다

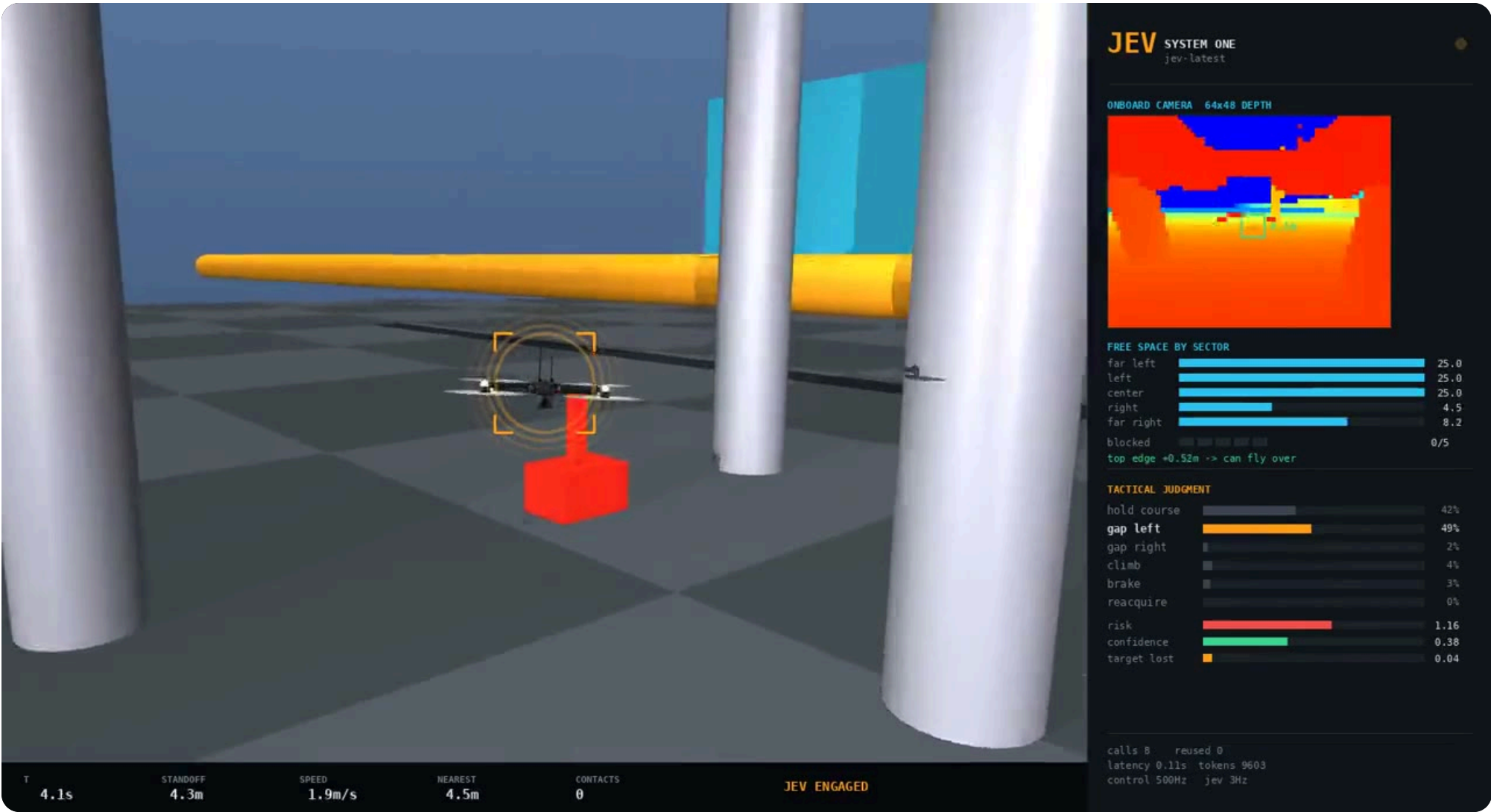


- Spot이 설비를 돌며 압력계, 사이트 글라스, 디지털 표시기를 찍으면 Gemini Robotics-ER 1.6이 값을 읽습니다.
- 읽는 방식: 사진을 확대하고, 바늘과 눈금에 점을 찍고, 코드를 실행해 비율을 계산합니다.
- 2026-04-08부터 Boston Dynamics 점검 제품의 가입 고객에게 적용됐습니다.

▶ 이미지를 누르면 영상 · Spot 설비 점검 (Boston Dynamics, 1분 40초)

AI는 장비를 조작하지 않고 계기를 읽기만 합니다. 판단은 사람이 합니다. 시험동 계기 순회 점검과 같은 종류의 일입니다.

드론은 비행 제어를 코드가, 전술 판단을 Jev가 맡습니다

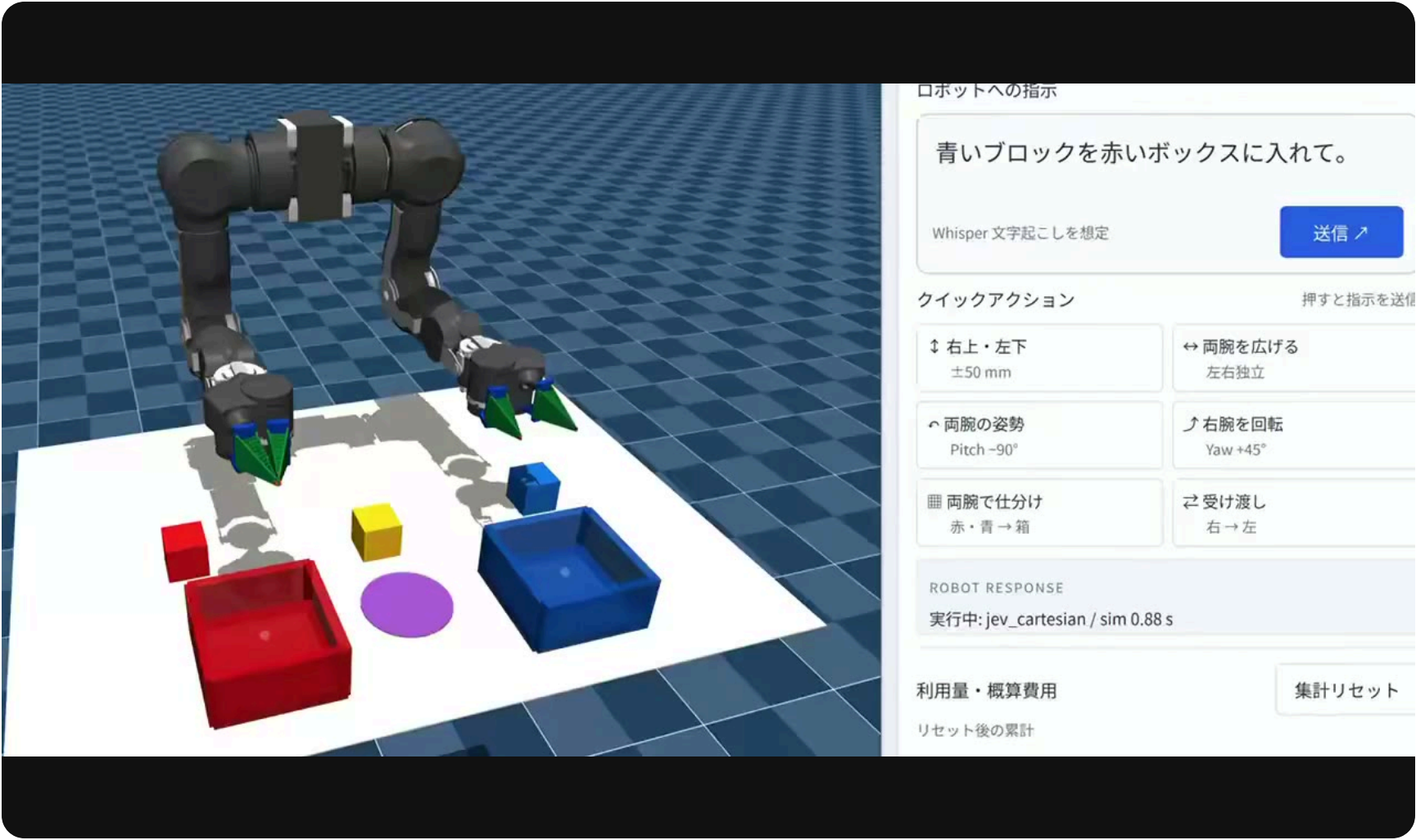


계층	누가	얼마나 자주
비행 제어·안전	코드	초당 약 500번
전술 판단	Jev	초당 약 2~3번

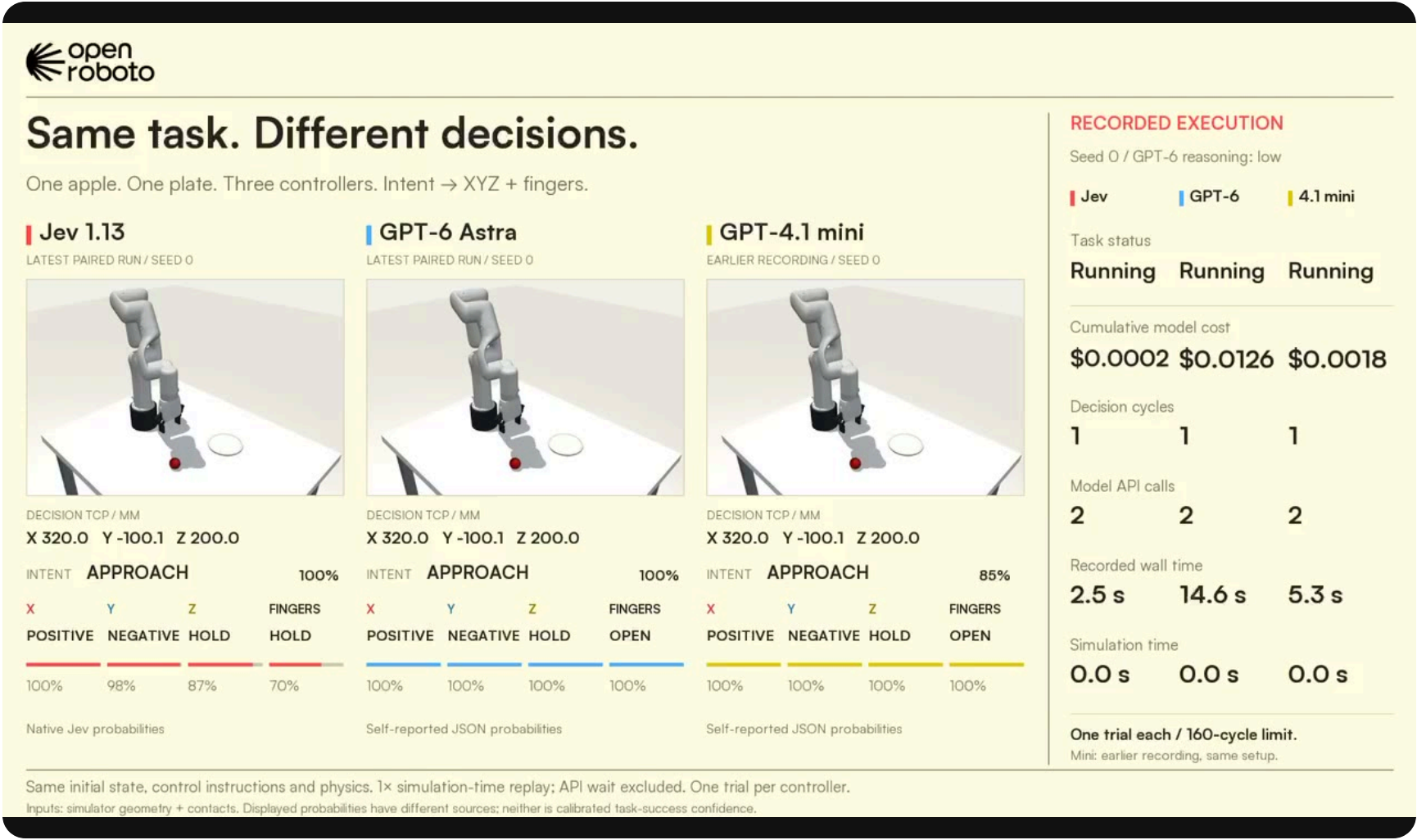
- Jev는 "앞의 장애물을 비켜야 하나" 같은 판단만 합니다.
- 응답이 70~500ms 걸리는 모델에게 실시간 제어를 맡기지 않습니다.
- 판단 주기와 제어 주기는 수백 배 차이가 납니다.

▶ 포스터를 누르면 영상이 열립니다. MuJoCo 물리 시뮬레이션 데모입니다. 주기 수치는 저자 보고 값이며 재측정하지 않았습니다.

로봇 팔도 미리 만든 동작 목록에서 고릅니다



▶ 양팔 로봇 · 원본 영상 열기 · "파란 블록을 빨간 상자에"라는 지시를 받으면 Jev가 다음 동작을 고르고, 역기구학과 물리는 코드가 계산합니다. MuJoCo 시뮬레이션입니다.



▶ 모델 비교 · 원본 영상 열기 · 같은 로봇 팔 과제를 Jev와 대화형 모델로 풀어 결정 시간과 비용을 비교한 게시자 영상입니다. 수치는 게시자 화면 기준입니다.

Q&A

오후에는 실습 폴더 C:\work\kimm 에서 이어집니다