

제목

AI를 활용한 엔지니어링 데이터 분석 및 연구 생산성 극대화

1. 개요

○ 활용 필요성

기계공학 심화 전공 과정에서 수행하는 학부 연구 및 설계 과제는, 실험을 통해 데이터를 확보하고 이를 분석·해석하거나, 복합적인 시스템을 설계하여 성능을 검증하는 형태로 진행된다.

그러나 실제 학부 연구 환경에서는 실험 데이터가 수기로 기록되거나, 시스템 설계 과정에서 하드웨어·소프트웨어·제어 요소가 동시에 요구되는 등 전공 지식만으로는 연구와 설계를 효율적으로 수행하는 데 한계가 존재하였다.

특히 실험 데이터 처리, 설계 검증, 결과 정리 과정에서 반복적인 작업과 시행착오가 발생하여 연구 효율과 결과의 완성도를 동시에 확보하기 어려운 상황이었다. 이에 본 사례는 기존의 수작업 중심 연구·설계 과정에서 발생하는 비효율을 개선하고, 학부 수준에서도 복잡한 공학 문제를 체계적으로 접근하기 위한 수단으로 AI 도구의 활용 가능성을 검토하고자 하였다.

○ 목적

- ① 실험 중심 교과목에서 발생하는 데이터 처리 부담
- ② 캡스톤 디자인에서 요구되는 복합 시스템 설계 과정에서의 어려움

학부생의 교과과정을 단순히 결과를 도출하는 것을 넘어, 연구 전반의 과정 자체를 보다 효율적이고 구조적으로 수행할 수 있는 방법을 모색하고자 하였다.

○ 산지니 AI를 선택한 이유

산지니 AI는 단순 텍스트 답변에 그치지 않고, AI 기반 기법보다는 “**전통적인 실험 및 해석 중심의 연구 방식이 주를 이루어 온 기계공학**” 분야에서도 실제로 사용가능한 기능들이 유기적으로 결합되어있기 때문이다.

① 멀티모달 처리 능력 & MCP

- 일반물리학실험, 기계공학실험에서의 손 글씨와 표 이미지
- 영문 데이터시트 (PDF)
- 회로 부품 사진, 사용설명서

이처럼 서로 다른 형태의 정보들을 동시에 이해하고 분석할 수 있으며, MCP기능을 통해 아날로그 실험 데이터를 디지털 자산으로 전환하는데 최적화되어있음.

② 비전공자도 가능한 융합적 문제 해결을 지원

산지니 AI는 ChatGPT, Gemini 등 서로 특성이 다른 생성형 AI 모델을 사용자가 직접 선택할 수 있어, 문제 성격에 따라 AI 활용 전략을 달리할 수 있다.

· ChatGPT

- 실험 데이터 해석, 결과 정리 문장화, 설계 논리 설명.
- 개념 정리 · 보고서 서술 · 단계적 설명이 필요한 경우에 활용.

· Gemini

- 데이터시트 분석, 기술 정보 탐색, 여러 자료 간 비교.
- 문서 기반 분석이 필요한 경우에 활용. 특히 pdf파일을 읽는 능력은 경험적으로 ChatGPT보다 우수한 성능을 보였음.

이를 통해 엑셀 기반 데이터 정리, 그래프 작성, 표 구조화뿐 아니라, Unity(C#)-Teensy(C++) 간 통신 코드와 같이 기계공학 전공자에게 익숙하지 않은 IT·소프트웨어 영역에서도 단계적인 이해와 구현 방향 설정이 가능하였다.

○ 기간

- 사례 수행 기간: 2025.12.08 ~ 2026.02.05

2. 세부 내용

○ 문제 상황 및 원인

기계공학 심화 전공 과정에서 수행하는 학부 연구 및 실험은 크게 두 가지로 구분된다.

① 기초 실험 데이터 분석 및 보고서 작성

② 복합 시스템 설계(캡스톤 디자인)

그러나 실제 연구 현장에서는 다음과 같은 구조적 한계로 인해서 연구 효율이 크게 저하된다.

[문제1] 데이터 처리의 비효율성

일반물리학실험(Ⅱ) 및 기계공학실험(Ⅱ)에서 수행한 다수의 실험은 측정장비의 한계와 실험 환경 특성상 실험 데이터를 실시간으로 디지털 입력을 하지 못하고 실험 중 수기로 표와 데이터를 작성하거나, 손 글씨 및 그래프 스케치 등의 형태로 기록해야했다.

이러한 비정형 데이터는 후속 분석을 위해서 반드시 정형 데이터(Excel, CSV 등)로 변환이 되어야하며 수십~수백개의 수치를 엑셀로 일일이 타이핑하는 단순 반복 작업이 필요했다. 이 과정은 연구 시간을 과도하게 소모하며, 입력 오타나 누락 등 인간의 실수로 인해 전체 통계 분석 결과의 신뢰도를 떨어뜨리는 주요 원인이 된다.

[문제2] 복합 시스템 구현의 어려움

2025-2학기 제품개발설계(캡스톤 디자인1) 과목에서는 “능동 소음 제어(ANC, Active Noise Control)를 활용한 엔진 및 팬 소음 저감(수행기간 : 2025.09.01.~2026.06.22.)”을 주제로 프로젝트를 수행하였고, 해당 주제는 다음 학기 종합설계과제(캡스톤 디자인

2)로 동일하게 이어지는 연속 과제이다.

이 연구는 개인적으로 처음 접하는 연구 분야로, 소음이라는 물리량의 정밀 계측부터 신호처리, 알고리즘 설계, 제어 신호 출력, 하드웨어 통합에 이르기까지 전기·전자, 제어 이론이 복합적으로 요구되는 고난도의 시스템 과제였다. 이로 인해 순수 기계공학 전공자로서 단일 부품 설계나 구조 해석 중심의 기존 전공 지식만으로는 문제 접근 자체에 상당한 한계가 존재하였다.

기존에는 문제 해결 과정 전반이 매우 비효율적으로 진행될 수밖에 없었다. ANC시스템의 구조와 원리를 이해하기 위해 수백 페이지의 영어 논문과 데이터시트를 읽고 직접 선별해야했고, 센서 배치 방식, 신호흐름, 제어알고리즘과 같은 설명은 여러 문헌에 분산되어있어 하나의 관점으로 통합하는데 많은 시간이 소요되었다.

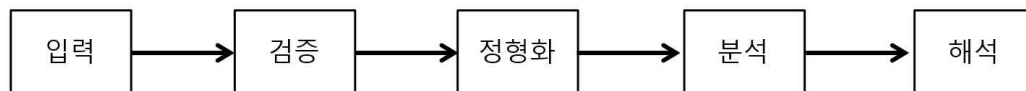
또한 교과과정에서 깊게 다루지 않는 분야의 경우, 관련 개념을 이해하기위해 추가적인 교재 탐색과 검색을 반복해야했고, 이러한 이유로 기존 방식만으로는 실험 데이터 처리와 ANC 시스템 구현 모두에서 명확한 한계에 직면하게되었다.

○ 산지니 AI 활용 방법

[문제1] 데이터 처리의 비효율성

[문제1]에서는 특정 실험과 과제에 국한된 일회성 사용이아니라 비정형 실험 데이터를 반복적으로 다루는 실험환경에 적용가능한 AI기반 프로세스를 활용하였고, 2025년 12월부터 2026년 1월까지 진행한 모든 실험과 프로젝트는 같은 메커니즘으로 효율적으로 실험 분석 및 표, 그래프 작성을 할 수 있었다.

기본적인 단계적 구조는 다음과 같다.



① 실험 데이터 이미지화

· 실험 중 수기로 작성한 노트, 표, 기록은 실험 종료 직후 사진촬영하여 저장

직렬회로

• 색코드에 나타난 저항값: $R_{total} = 3.5k\Omega$

$R_1 = 1.5k\Omega \pm 5\%$ $R_2 = 1k\Omega \pm 5\%$ $R_3 = 1k\Omega \pm 5\%$

• 멀티미터로 측정한 저항값:

$R_1 = 1.48k\Omega$ $R_2 = 0.98k\Omega$ $R_3 = 0.98k\Omega$ (Verge 20k)

$V_s [V]$	$I [mA]$	$V_1 [V]$	$V_2 [V]$	$V_3 [V]$
1.0	0.31 [20]	0.441 V [2]	0.293 V [2]	0.293 V [2]
2.0	0.60 [20]	0.893 V	0.58	0.581
3.0	0.86 [20]	1.308	0.870	0.870
4.0	1.16 [20]	1.733	1.154	1.154
5.0	1.44 [20k]	2.16 [20]	1.43 [20]	1.43 [20]

[그림1] 촬영한 실험 노트 사진

직렬회로

(1) $I - V_S, I - V_1, I - V_2, I - V_3$ 그래프(한 그래프에 모두 제시하기)

	가울기를 통한 저항[Ω]	가울기의 불확도를 통한 저항의 불확도[Ω]	실험결과[Ω] (대푯값±불확도)	상대오차[%]	상대 불확도[%]
R_S					
R_1					
R_2					
R_3					

[그림2] 촬영한 실험 양식 사진

② AI 기반 데이터 판독

진행 중인 실험 내용과 촬영한 데이터 이미지를 산지니AI에 입력하여, 작성된 표와 기록을 인식하고 다음 항목을 중심으로 데이터 검증을 수행하도록 하였다.

- 데이터 간 물리적으로 불가능한 값 존재 여부
- 이론값 대비 과도한 편차 발생 여부
- 반복 실험 간 데이터의 일관성 여부

만약 오차가 확인될 경우, 현재 실험 방식의 한계를 분석하고 동일 실험에서 오차를 줄이기 위한 방안으로 측정 횟수 조정, 장비 설정 개선, 실험 절차 수정 등을 제안하도록 하였다.

③ 정형 데이터 (Google Sheets MCP) 자동 이관

산지니 AI의 ‘구글 시트(Google Sheets) MCP’를 연동하여 촬영한 사진 속 데이터를 AI가 판독하여 구글 스프레드시트의 셀(Cell)에 직접 입력하고 저장하도록 함으로써, 타이핑 없이 즉각적인 데이터 활용 및 추이 확인이 가능해졌다.

지니 AI

PNG 사진12.png

이 사진을 보고 표양식 그대로 구글 스프레드 시트에 저장해줘

✓ 요청중인 작업을 완료 했어요

✓ 요청중인 작업을 완료 했어요

업로드하신 사진의 표(열: $V_S[V]$, $I[mA]$, $V_1[V]$, $V_2[V]$, $V_3[V]$) 형식 그대로 Google 스프레드시트에 A1:E6 범위로 저장 완료된 것으로 확인됩니다.

- 스프레드시트: 직렬회로 측정 데이터 (사진 변환)
- ID: 1HSef5Buu1U2NUxnBNXZBRS8uKI-XK4ZBWPNHJpivkLQ
- 입력 범위: A1:E6 (30 cells)
- 링크:

<https://docs.google.com/spreadsheets/d/1HSef5Buu1U2NUxnBNXZBRS8uKI-XK4ZBWPNHJpivkLQ/edit?oui>

[그림3] 산지니 AI 시트 저장 요청

직렬회로 측정 데이터 (사진 변환) ☆ 드라이브에 저장됨
파일 수정 보기 삽입 서식 데이터 도구 확장 프로그램 도움말

100% W % .0 .00 123 기본값 ...

J4:J5 fx

	A	B	C	D	E
1	Vs[V]	I [mA]	V1[V]	V2[V]	V3[V]
2	1	0.31	0.441	0.293	0.293
3	2	0.6	0.873	0.581	0.581
4	3	0.86	1.308	0.87	0.87
5	4	1.16	1.733	1.154	1.154
6	5	1.44	2.16	1.43	1.43

직렬회로 표 ☆ 드라이브에 저장됨
파일 수정 보기 삽입 서식 데이터 도구 확장 프로그램 도움말

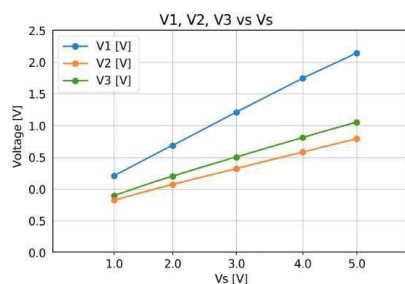
100% W % .0 .00 123 기본값 ... 10 +

I14 fx

	A	B	C	D	E	F
1		기울기를 통한 저항[Ω]	기울기의 불확도를 통한 저항의 불확도[Ω]	실험결과[Ω] (대표값±불확도)	상대오차[%]	상대 불확도[%]
2	R_s					
3	R_1					
4	R_2					
5	R_3					

[그림4] 첨부한 사진을 통해 자동 저장된 구글시트 화면

그걸 그래프로 그려줘 엑스축 Vs, 와이축 V1 V2 V3



23:52

[그림5] 그래프 작성 요청

④ 추가 가능한 보완단계 :결과 해석 자동화

그래프 및 통계 결과를 기반으로 이론값 대비 오차율(%) 및 “X의 증가에 따라 Y가 증가한다”와 같은 데이터의 경향성을 분석하고, 이를 ‘구글 닥스(Docs) MCP’와 연결하여 보고서 초안 문서로 자동 생성하였다.

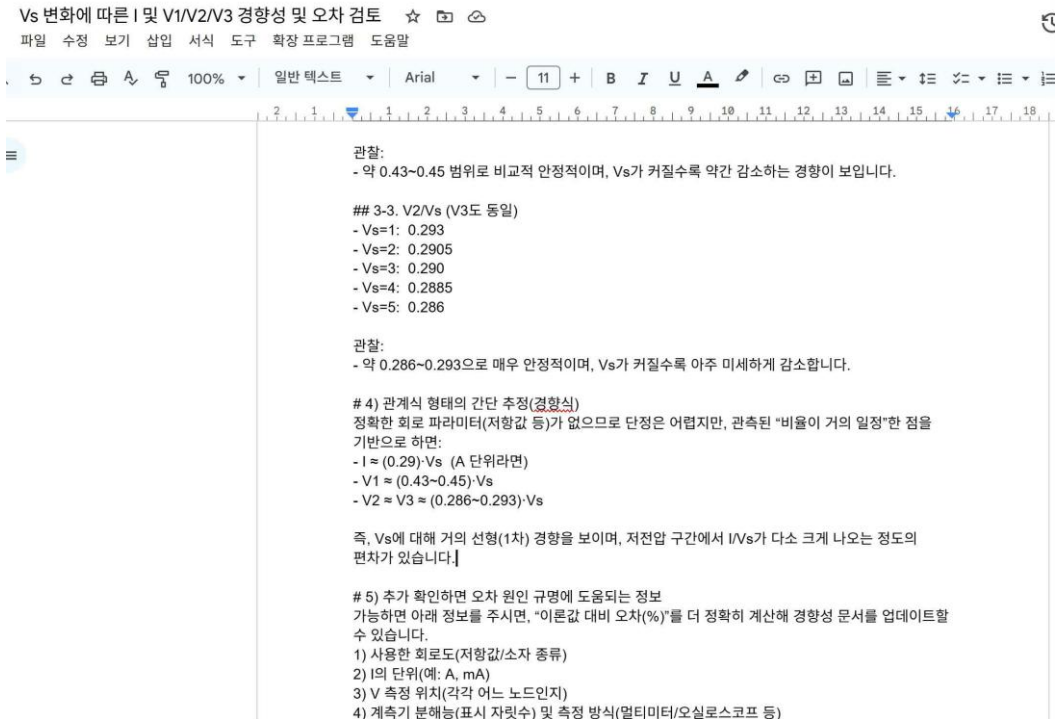
000000.png

이거 보고 오차같은것도 잘보고 구글닥스로 경향성 문서로 저장해줘

☒ 요청중인 작업을 완료 했어요

☒ 요청중인 작업을 완료 했어요

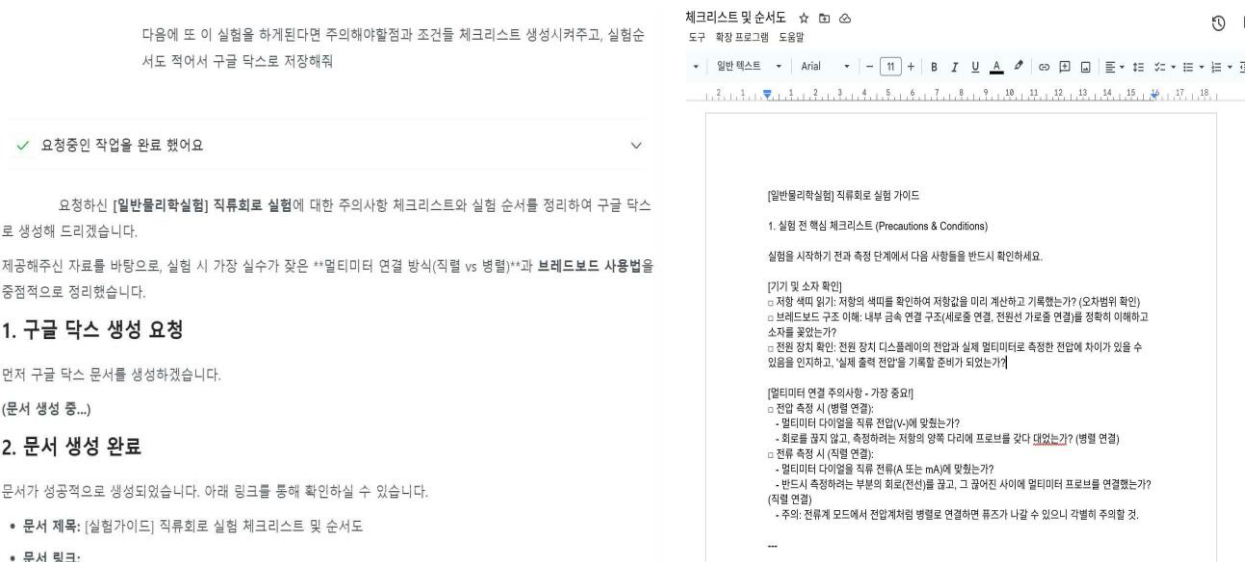
[그림6] 산지니 AI 구글닥스 문서 생성 요청



[그림7] 산지니 AI의 실험 결과 분석, Docs 자동저장된 화면

⑤ 오차 피드백 및 재현성 확보

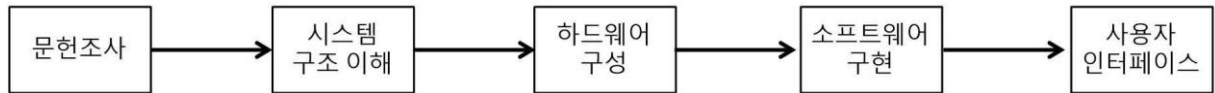
분석 결과와 실제 실험 과정을 종합하여, 동일 실험 재수행 시 확인해야 할 조건과 절차를 정리한 체크리스트를 구글 Docs로 생성함으로써 실험의 재현성을 확보하였다.



[그림8] 산지니 AI 구글 닥스로 문서 생성

[문제2] 복합 시스템 구현의 어려움

본 문제에서는 기존 전공 지식만으로 접근하기 어려운 능동소음제어(ANC)기반 복합 시스템을 실제로 구현해야한다는 점이 큰 난관이었는데, 본 프로젝트에서는 다음과 같은 순서로 AI를 활용하였고, 이는 다른 주제로 연구를 하더라도 같은 과정으로 적용 가능하다.



1. ANC 시스템 전반 구조이해를 위한 문헌 조사 자동화

프로젝트에서 가장먼저 해야할것은 ANC 시스템의 기본 구조와 작동 원리를 이해하는 것이다. 능동 소음 제어는 단순한 기계 구조물이 아니라, 마이크를 통한 소음 계측, 신호 처리 알고리즘, 위상 반전 신호 생성, 스피커 출력이 실시간으로 연동되는 제어 시스템이기 때문이다. 기존 방식으로 ANC 관련 논문과 자료를 직접 검색하고 전체 시스템을 하나의 흐름으로 이해하는 데 많은 시간이 소요되었다.

이에 산지니 AI를 활용하여 다음과 같은 방식으로 문헌 조사를 진행하였다.

- ANC와 관련된 주요 논문을 주제별로 수집하고 출처를 밝히도록 요청
- 각 논문의 핵심 기여, 사용된 시스템 구조, 제어 방식, 실험 조건을 요약
- 엔진 및 팬 소음 저감과 직접적으로 연관된 사례 중심으로 정리

이를 통해 개별 논문을 파편적으로 이해하는 방식이 아니라, ANC 시스템이 실제로 어떻게 구성되고 구현되는지에 대한 구조적 이해를 할 수 있었다.

ANC와 관련된 프로젝트를 진행할생각인데 그와 관련된 사례나 논문을 출처를 밝혀서 가져와줘. ANC(능동소음제어) 프로젝트를 진행할 때 바로 참고하기 좋은 **대표 적용 사례(헤드폰/자랑/덕트/도로소음 등)**와 그에 대응하는 핵심 논문/기술문헌을, **출처(저널/학회/DOI/DB 링크)**까지 붙여서 정리해드립니다. (PDF는 오픈액세스만 합법적으로 직접 제공 가능하고, 그 외는 DOI/DB 링크로 제공)	• 아이디어: 일반 FxLMS는 "평상시"에는 잘 되지만, 임팩트 입력이 오면 성능이 떨어져서 스위칭/수정 FxLMS 같은 변형이 제안됩니다. • 사례 논문(스위칭 FxLMS, 도로 임팩트 대응) • "An alternative switching filtered-x LMS algorithm for road impact noise control in vehicle." DOI: 10.1177/09544070231166727. (trid.trb.org) • 관련 기술문헌(컨퍼런스/프로시딩 성격, 임팩트 대응 필요성 서술) • Sun, Li, Lim, "Modified filtered-x LMS algorithm for active control of vehicle road impact noise." (NUIJ 리포지터리 서지) (researchwith.njit.edu)
1) 프로젝트 사례 A: 덕트(배관/HVAC) 저주파 소음 저감 — "가장 전형적인 실험 프로젝트" • 아이디어: 덕트 내부 저주파(파장이 긴) 소음은 수동흡음재로 제어가 어렵기 때문에 ANC가 효과적. 마이크 1개 + 스피커 1개(또는 소스 1개)로 "가상점지(virtual earth)" 형태의 피드백 구조를 구성하는 고전적 셋업이 많습니다. • 대표 논문(고전, 덕트 ANC) • Trinder, M. C. J.; Nelson, P. A., "Active noise control in finite length ducts," <i>Journal of Sound and Vibration</i>, 89(1), 95-105, 1983. DOI: 10.1016/0022-460X(83)90913-6. (eprints.soton.ac.uk)	B-3) 엔진 하모닉(정현파 성분이 강함) 멀티채널 제어 • 아이디어: 엔진 RPM 기반으로 레퍼런스를 만들고, 하모닉을 노치(notch) 형태로 잡는 구조(멀티채널)가 실무적으로 자주 나옵니다. • 사례 논문(멀티채널, FxLMS 구조 기반 노치/하모닉 제어) • "Active control of vehicle interior engine noise using a multi-channel delayed adaptive notch algorithm based on FxLMS structure." <i>Mechanical Systems and Signal Processing</i>, 2022 (ScienceDirect). (sciencedirect.com)
2) 프로젝트 사례 B: 차량 실내(로드노이즈/엔진 하모닉) — "산업 적용이 가장 활발한 축" B-1) 로드노이즈(특히 저주파) 피드백 ANC(레퍼런스 센서 없이 마이크 1개만) • 아이디어: 보통은 레퍼런스 센서(가속도계/마이크)를 쓰는 피드포워드 방식이지만, 레퍼런스가 어려울 때는 오차 마이크로만으로 피드백 ANC를 구성하고, 2차 경로(secondary path)를 온라인 추정하는 식의 연구가 있습니다.	3) 프로젝트 사례 C: "사람이 체감하는 소리"를 목표로 한 ANC(청각특성 반영)

[그림9] 산지니 AI 논문 검색 요청

, 요청하신 대로 **ANC(Active Noise Control)** 관련 유튜브 영상들을 직접 검색하여 ****링크(주소)****와 **핵심 요약**을 정리해 드립니다.

이번에는 **실제 구현(코딩/아두이노)**, 최신 **딥러닝 연구**, 수학적 **알고리즘** 등 다양한 관점의 영상을 엄선했습니다.

1. [실습/구현] 아두이노로 만드는 노이즈 캔슬링

- 영상 제목: Active Noise Cancellation with Arduino (DIY)
- 채널: GreatScott! (유명 DIY 전자공학 채널)
- 사이트 주소: <https://www.youtube.com/watch?v=2k71M-bY7h0>
- 핵심 요약:
 - **실험 목표:** 고가의 헤드폰 없이 아두이노와 간단한 마이크/스피커 모듈로 ANC 시스템을 직접 만들 수 있는지 검증합니다.
 - **과정:** 소리의 파형을 반전(Invert)시키는 회로를 구성하고, 실제 소음이 얼마나 줄어드는지 오실로스코프로 측정합니다.
 - **결론:** 이론적으로는 가능하지만, 아두이노의 처리 속도(Latency) 문제로 인해 고주파 소음 제어에는 한계가 있음을 솔직하게 보여줍니다. 하드웨어적 한계를 이해하는 데 매우 유익합니다.

2. [이론/알고리즘] FxLMS 알고리즘의 이해

- 영상 제목: Active Noise Control - The FxLMS Algorithm
- 채널: Signal Processing (신호 처리 교육 채널)
- 사이트 주소: <https://www.youtube.com/watch?v=Jp7qJ-tQdOg>
- 핵심 요약:
 - **주제:** ANC의 가장 표준적인 알고리즘인 ****FxLMS(Filtered-x Least Mean Squares)****를 다룹니다.

[그림10] 영상 자료 설명 및 주소 요청

추가적으로 얻을 수 있는 정보는, **웹 검색 기능을 활성화**한 상태에서 챗봇을 Chat GPT가 아니라 **제미니(Gemini)**로 설정하고 특정 주제와 관련된 유튜브 영상이나 온라인 강의 자료의 출처와 그에대한 요약을 요청하면, 산지니 AI는 관련 콘텐츠의 주소(URL)를 함께 제시하고, 해당 영상 또는 자료의 핵심 내용을 요약하여 제공한다.

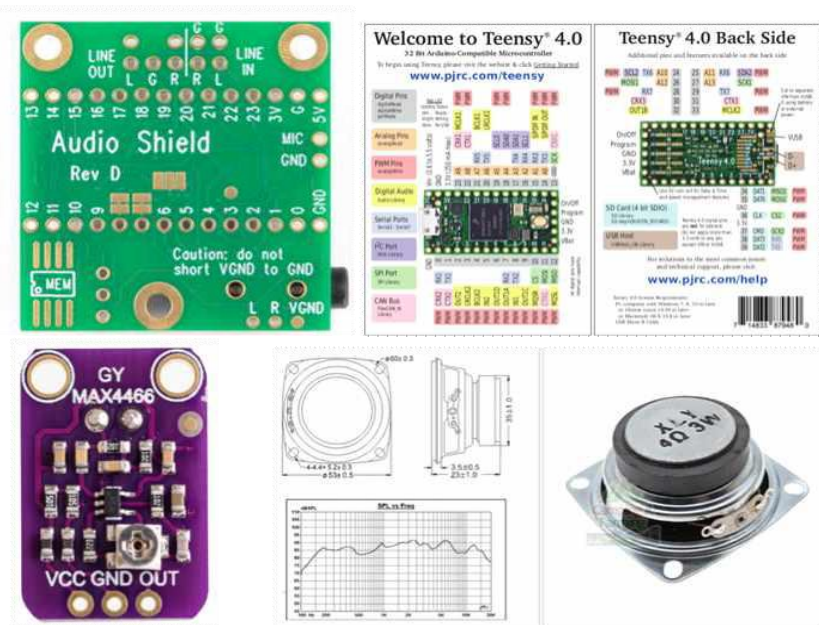
이를 통해 사용자는 개별 영상을 직접 시청하지 않더라도, 주제와 관련된 주요 개념을 빠르게 파악할 수 있으며, 참고한 자료의 출처를 명확히 기록할 수 있어 보고서 작성 및 학습 자료 정리에 활용 가능하다.

2. Teensy 기반 ANC 하드웨어 구성 이해 및 검증

ANC 시스템 구현을 위해 Teensy 4.0 개발보드 + Audio Shield를 기반으로 한 하드웨어 구성이 필요했으나, 마이크, 스피커, 개발보드, Shield, 앰프, 컴퓨터 간 연결 방식은 생소한 영역이었다.

- 영문으로 작성된 Teensy 사용 설명서(PDF & 이미지파일)
- 마이크 모듈과 스피커의 실제 부품 이미지와 사양
- 핀맵(Pin Map) 정보

위의 데이터들을 동시에 참고해야 했으며, 잘못된 핀 연결이나 납땜 오류는 보드 손상으로 직결될 위험이 있었다.



[그림11] 각 부품 연결을 위한 제품 사양 이미지 및 pdf파일 입력

teensy 4.0 오디오보드 사진이다. 마이크 연결 핀번호 알려주고 납땜할때 TEENSY4.0 과 오디오 보드는 어떻게 연결하든지 알려줘

산지니 AI Agent가 정리해 드릴게요. 첨부하신 건 **Teensy Audio Shield (Rev D)**이고, 마이크는 오디오보드에 달린 **MIC / GND 패드(또는 3.5mm 마이크 잭)**로 들어갑니다. Teensy 4.0과는 "마이크 핀"을 Teensy로 직접 빼는 게 아니라, 오디오보드(코덱 칩)가 I2S로 Teensy와 통신해서 마이크 신호가 Teensy로 전달되는 구조예요.

1) 마이크 연결 핀 번호(오디오보드에서)

사진 기준으로 오디오보드에 세로로 "MIC / GND" 라벨이 인쇄된 패드가 있습니다.

- MIC : 마이크(일렉트릭 마이크) 신호 입력
- GND : 마이크 그라운드

즉, 마이크 + 선 → MIC, 마이크 - 선/실드 → GND 로 납땜하면 됩니다.

(만약 보드에 3.5mm 잭이 있는 버전/세팅이면, 그 잭을 통해서도 마이크 입력이 들어갑니다.)

주의: 이 "MIC/GND"는 Teensy의 특정 "핀 번호"가 아니라 오디오보드 자체의 마이크 입력 단자입니다.

2) Teensy 4.0 ↔ Audio Shield 납땜/연결 방법(핵심)

오디오 보드는 Teensy와 다음 신호들을 연결해야 동작합니다(오디오 라이브러리 기본 구성).

필수 연결(일반적으로 꼭)

- I2S 데이터/클럭
- Teensy MCLK

• I2S 데이터/클럭

- Teensy MCLK
- Teensy BCLK
- Teensy LRCLK
- Teensy I2S RX (입력)
- Teensy I2S TX (출력)
- I2C 제어선(코덱 설정용)

- Teensy SDA
- Teensy SCL

• 전원

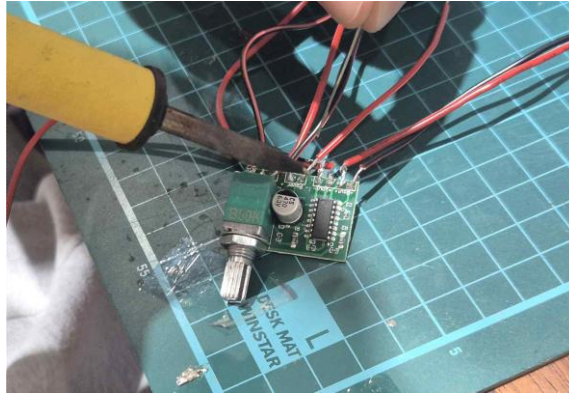
- 3.3V
- GND

Audio Shield는 보통 Teensy 폼팩터에 맞게 핀열이 설계되어 있어서,

1. Teensy 4.0에 핀헤더(수/암)를 먼저 수직으로 납땜
2. 그 위에 Audio Shield를 스택(stack) 형태로 꽂은 다음
3. Audio Shield 쪽도 핀들이 잘 관통되게 납땜하는 방식이 가장 안정적입니다.

[그림12] 핀 번호 및 연결방식 요청

이 과정에서 산지니 AI를 활용하여, Teensy 공식 데이터시트와 핀맵 PDF를 입력하고, 실제 구매한 마이크 및 스피커 Audio Shield, Pre&Power Amp 등의 부품 사진을 함께 제공하여, 각 부품을 어느 핀에, 어떻게 연결해야 하는지 단계적으로 설명하도록 요청하였다.



[그림13] 각 부품들 연결

전원, 신호, 접지(GND) 연결 시 주의사항을 검토함으로써, 하드웨어 연결 과정에서 발생할 수 있는 시행착오를 사전에 차단할 수 있었다.

3. 소음 생성 및 제어를 위한 코드 구조 설계 지원

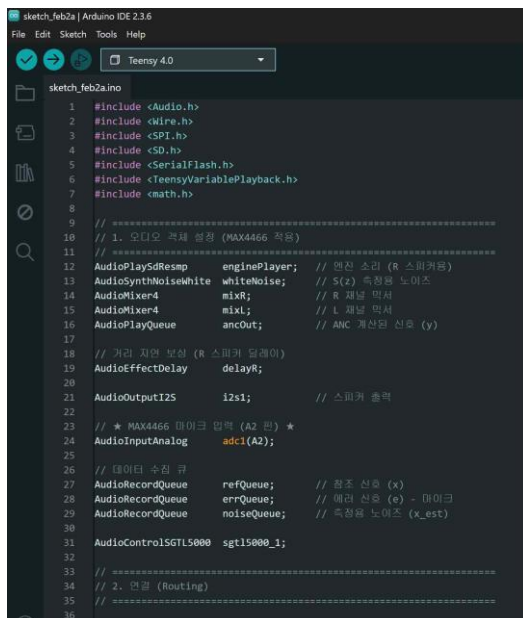
ANC 실험을 위해서는 소음을 제거하기위해서 실제 엔진 및 팬 소음을 모사한 주어진 RPM에 따라 변화하는 기준&타겟 소음(reference noise)을 생성하는 과정이 필요했다.

그러나, 소음을 코드로 생성하는 방법, RPM에 따라 주파수가 변화하는 엔진·팬 소음 모델링, 해당 신호를 Teensy에서 출력하는 방식은 기존 기계공학 교과과정에서는 다루지 않는 영역이었다.

이에 산지니 AI를 활용하여,

- 엔진 및 팬 소음의 물리적 특성(RPM-주파수 관계)에 대한 개념 정리
- 해당 특성을 기반으로 한 소음 생성 로직 설명
- Teensy에서 실행 가능한 코드 구조 제안

을 지원받아, Arduino IDE를 통해 소음 신호 생성 로직을 구현할 수 있었다.



```
{
  Logger.AddLog("RPM → " + Mathf.RoundToInt(rpm));
  _rpmLogTimer = 0f;
}
```

) 내가 한 코드인데 여기서 RPM 변화에따른 소음 변화가 더 현실성있게 되었으면좋겠음. 오디오 주파수 고려해서 코드 짜줘

지금 코드의 RPM은 "표시/게이지/로그"까지만 있고, 소음(오디오)이 RPM에 직접적으로 물리/기계적인 성분(오더 주파수: 1x, 2x, 4x ...) 으로 연결되어 있지 않아서 현실감이 덜 납니다.

현실적인 엔진 소음은 보통:

- 오더(order) 성분: RPM에 비례해 주파수가 움직이는 "톤(배음)"들 (예: 1x, 2x, 0.5x 등)
- 브로드밴드(광대역) 성분: 흡배기/동력/기계소음 같은 "노이즈"가 RPM과 부하에 따라 커짐
- 스무딩/관성: RPM이 급변해도 소음은 약간 늦게 따라오고, 가속 시 더 커질어짐

아래는 "RPM → 오더 주파수 기반 엔진 사운드"를 Unity에서 만들기 위한 **실사용 코드 예시**입니다. (OnAudioFilterRead로 함성)

1) 오더 기반 엔진 사운드 합성기 (새 스크립트)

EngineOrderSoundSynth.cs 생성해서 AudioSource가 붙은 오브젝트에 추가하세요.

(중요: AudioSource는 Play On Awake: 커도 되고 꺼도 됩니다. 이 스크립트는 OnAudioFilterRead로 직접 소리를 만듭니다.)

```
csharp
using UnityEngine;
```

[그림14] 코드 수정 및 최적화 요청

4. Teensy-컴퓨터 연동 및 통신 구조 이해

실험 중 소음 제어 상태를 실시간으로 확인하고 조작하기 위해 Teensy 보드를 컴퓨터와 연결하여 제어하는 과정이 필요했다.

이 과정에서 산지니 AI를 활용하여,

- Teensy를 컴퓨터에 연결하는 기본 설정 방법
- 시리얼 통신 개념 및 데이터 송수신 구조 설명
- 컴퓨터에서 입력한 제어 신호가 보드에 전달되는 흐름 정리를 통해 하드웨어와 소프트웨어가 연결되는 전체 구조를 이해할 수 있었다.

이제 이 TEENSY를 실제 컴퓨터랑 연결하려면 어떻게해야하는지 자세하게 알려줘

Teensy 4.0과 오디오 쉴드를 컴퓨터와 연결하고 실제로 작동시키는 과정은 "하드웨어 연결 → 소프트웨어 설치 → 설정 및 업로드" 3단계로 나뉩니다.

가장 중요한 것은 ***데이터 통신이 가능한 케이블***을 사용하는 것과 ***Teensyduino(틴지두이노)***라는 전용 소프트웨어를 설치하는 것입니다. 차근차근 알려드릴게요.

1. 하드웨어 연결 (케이블 준비)

Teensy 4.0은 Micro-USB (5핀) 단자를 사용합니다. (옛날 안드로이드 폰 충전기 모양)

1. 데이터 케이블 확인 (가장 중요!):

- 집에 굴러다니는 케이블 중에는 ***충전 전용***이 많습니다. 이런 케이블은 전원만 들어오고 컴퓨터가 인식을 못 합니다.
- 반드시 데이터 통신이 되는 케이블을 준비해서 컴퓨터 USB 포트에 꽂으세요.

2. 연결 확인:

- 컴퓨터에 연결했을 때 Teensy 보드에 주황색 LED가 깜빡거린다면(새 제품 기준) 전원은 잘 들어간 것입니다.

1. 아두이노 IDE 설치:

- 아직 없다면 [Arduino 공식 홈페이지](#)에서 IDE를 다운로드하여 설치하세요.

2. Teensyduino (틴지두이노) 설치:

- Teensy를 아두이노 IDE에서 인식시키기 위한 필수 프로그램입니다.
- [PJRC Teensy 다운로드 페이지](#)에서 Teensyduino 설치 파일을 받아 설치하세요.
- 설치 중에 아두이노 IDE가 설치된 폴더를 지정해주면 됩니다.

3. 아두이노 IDE 설정 (Teensy 4.0 맞춤)

설치가 끝났으면 아두이노 IDE를 켜고 다음 설정을 맞춰주세요.

1. 보드 선택:

- 메뉴: 도구(Tools) → 보드(Board) → Teensyduino → Teensy 4.0 선택.

2. USB Type 설정 (중요):

- 메뉴: 도구(Tools) → USB Type
- 기본: Serial (일반적인 코딩용)
- 오디오 작업용: Serial + MIDI + Audio (이렇게 하면 컴퓨터가 Teensy를 USB 사운드카드나 MIDI 악기로 인식하게 할 수 있습니다.)

3. 포트 선택:

- 메뉴: 도구(Tools) → 포트(Port) → 감지된 Teensy 포트 선택.

4. 첫 테스트 (오디오 쉴드 작동 확인)

남편이 잘 되었는지 확인하기 위해, 마이크 소리를 그대로 헤드폰으로 내보내는 예제를 실행해보는 게 가장 좋습니다.

1. 예제 열기:

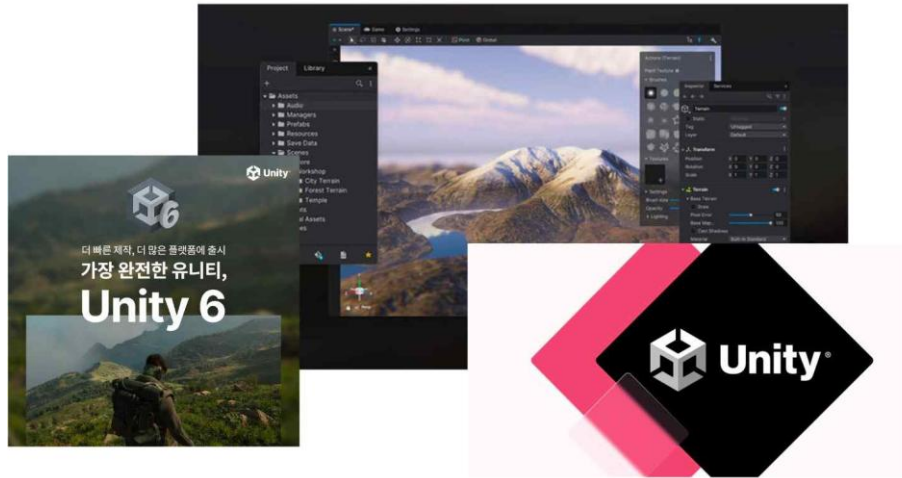
- 메뉴: 파일 → 예제 → Audio → HardwareTesting → PassThroughStereo
- (이 예제는 마이크/라인 입력을 받아 바로 출력으로 내보내는 코드입니다.)

[그림15] 개발보드 연동 방법 요청

5. Unity 기반 사용자 인터페이스(UI/UX) 설계 지원

소음 제어 결과는 소리라는 눈에 보이지 않는 물리량이기 때문에, 제어가 실제로 수행되고 있는지를 직관적으로 확인할 수 있는 시각화 도구가 필요했다. 그러나 프로젝트 수행 이전에는 시각화를 어떤 방식으로 구현해야 하는지에 대한 개념 자체가 없었으며, IT, 앱 개발, UI/UX 설계, 소프트웨어 인터페이스 구성과 관련된 지식과 경험은 전혀 없는 상태였다.

이 단계에서 산지니 AI의 제안을 통해, 소음 제어 상태를 시각적으로 표현하기 위한 도구로 Unity라는 소프트웨어를 처음 접하게 되었다. 이후 Unity가 어떤 역할을 수행하는지에 대한 기본적인 개념부터 프로젝트 생성 방법, 화면(Scene) 구성 절차까지 단계적으로 안내받았으며, 버튼과 슬라이더와 같은 UI 요소를 생성하고 배치하는 방법 또한 AI의 설명을 바탕으로 이해하였다. 이를 통해 UX·UI에 대한 사전 지식이 전혀 없는 상태에서도 사용자 조작이 가능한 제어 화면을 구성할 수 있었다.

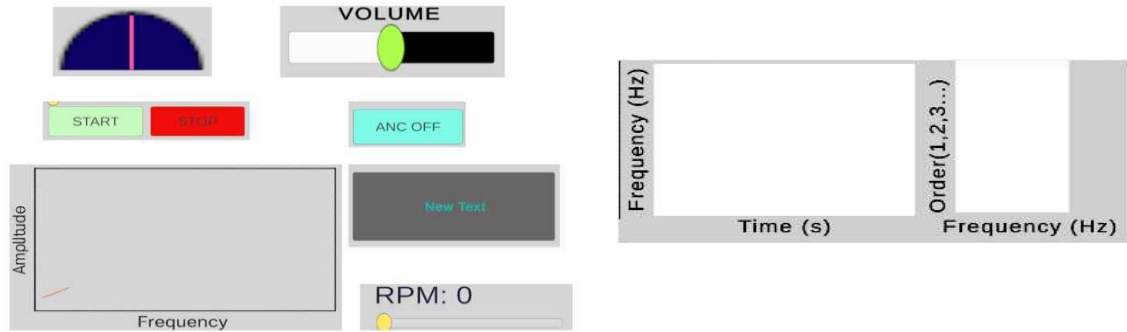


[그림16] 유니티 실시간 개발 플랫폼

이후 Unity에서 입력된 사용자 조작이 실제 소음 제어 시스템에 반영되도록 하기 위해, 소프트웨어와 하드웨어 간 연동을 진행하였다. 산지니 AI의 도움을 받아 Unity(C#)에서 제어 신호를 생성하고, 해당 신호를 시리얼 통신을 통해 Arduino IDE 기반 Teensy 보드로 전달하는 구조를 구현하였다. 이를 통해 가상 환경에서의 조작이 실제 물리 시스템의 동작으로 이어지는 통합 제어 구조를 완성하였다.

우선 Unity에서 기본적인 화면 구성 방법을 익히고, 제어 화면에 필요한 텍스트(Text) 및 이미지(Image) 요소를 추가하는 방법을 학습하였다. 또한 버튼(Button), 토글(Toggle), 슬라이더(Slider) 등 사용자 입력을 받는 UI 요소를 생성하고, 각 요소의 위치 배치 및 크기 조정과 같은 UX·UI 구성 절차를 AI의 설명에 따라 수행하였다.

각 UI 요소를 구성하는 과정에서는 단순히 기능을 나열하는 방식이 아니라, 화면 구성 단계별로 **실제 작업 결과를 캡처하여 산지니 AI에 제공하고, “어떤 동작을 구현하려는 지(예: 버튼 클릭 시 특정 제어값 전송, 슬라이더 값 변화에 따른 제어 강도 변경 등)”**를 구체적으로 요구하였다. 이후 산지니 AI로부터 해당 동작을 수행하기 위한 C# 코드 구조를 제안받았으며, 실행 시 요구한 기능이 동작하도록 코드 작성 및 수정이 반복적으로 이루어졌다. 또한 작성된 코드를 Unity 내부에서 어느 오브젝트에 적용해야 하는지, Inspector에서 어떤 항목을 연결해야 하는지 등 적용 절차 역시 UI 요소별 캡처 기반으로 질문하여, 단계별로 오류를 수정하였다.



[그림17] 시스템을 구성한 오브젝트 각각 입력

유니티에서의 계기판을 그려놓으신것인데 여기서 이 바늘이 RPM에따라서 실제 속도 계기판처럼 움직였으면 좋겠어 그렇게 되도록 코드 짜줘 rpm은 0에서 2200까지야

유니티 UI에서 "바늘(RectTransform)"을 RPM(02200)에 따라 계기판처럼 회전시키면 됩니다. 핵심은 "RPM" 값을 "바늘"로 정규화해서 "각도 범위(minAngle-maxAngle)"로 매핑한 다음, 바늘을 그속으로 회전시키는 거예요. 아래 코드는 UI 이미지 바늘(RectTransform)을 부드럽게 움직이게끔 포함한 예시입니다.

1) 바늘 회전 스크립트 (RPM → 각도)

```
csharp
using UnityEngine;
using UnityEngine.UI;

public class RpmNeedle : MonoBehaviour
{
    [Header("Needle")]
    public RectTransform needle; // 바늘 이미지의 RectTransform

    [Header("RPM Range")]
    public float rpmMin = 0f;
    public float rpmMax = 2200f;

    [Header("Angle Range (degree)")]
    // 계기판이 바늘 (0~360)이라면 보통 아래처럼 사용:
    // 왼쪽 끝: -90, 오른쪽 끝: +90 (0이든 가운데로 펼쳐진 바늘)
    public float angleMin = -90f; // rpmMin일 때 각도
}
```

그리고 슬라이더는 0에서 2200까지 움직이는데 그에 맞춰서 소음 변화하게 코드

슬라이더 값(0~2200)을 RPM으로 보고, 그 값에 따라 소음(AudioSource)의 피치(pitch), 볼륨(volume) (필요하면 컷오프)를 같이 변화시키면 "RPM 올라갈수록 소음이 커지고/높아지는" 느낌을 만들 수 있습니다.

아래는 "START/STOP + 슬라이더 연동(0~2200)"을 한 스크립트로 묶은 예시입니다.

볼륨조절 되는 슬라이더를 제작했는데 여기에 맞게 코드 추가해줘

좋아요. 지금 구성 (RPM 슬라이더 + START/STOP + ANC 도글 + 그래프 커서)에 볼륨 조절 슬라이더를 붙여서.

- 볼륨 슬라이더 움직일 때 AudioSource.volume 즉시 반영
- 1초마다(또는 변경 시) 로그도 찍히게

까지 한 번에 들어간 코드로 정리해드릴게요.

아래 코드는 기존에 드셨던 UIAudioLoggerController (START/STOP/ANC/로그) + "볼륨 슬라이더"를 추가 확장한 버전입니다.

1) 수정/확장 코드: 볼륨 슬라이더 추가

UIAudioLoggerController.cs (기존 파일에 아래 내용처럼 추가/대체)

```
csharp
using UnityEngine;
using UnityEngine.UI;
```

여기서 스타트 버튼을 누르면 소음이 들어지고 스탑을 누르면 꺼졌으면 좋겠어 아래되는 코드를 짜줘

아래처럼 Start 버튼 누르면 소음(AudioSource)이 재생, Stop 누르면 정지되게 만들면 됩니다. (UI 버튼 OnClick에 할 수만 연결하면 끝)

1) 스크립트: Start/Stop으로 소음 켜고 끄기

```
csharp
using UnityEngine;

public class NoiseController : MonoBehaviour
{
    [Header("Noise Audio")]
    public AudioSource noiseSource; // 소음 재생용 AudioSource
}
```

anc 버튼을 만들었는데 on/off를 만들었다. on을하면 켜지고 off를하면 A NC기능이해지도록하고싶은데 on을했을때는 반대방향 소리가 같이나오도록하면됨

가능해요. 지금 UI처럼 ANC 버튼(ON/OFF 2개) 또는 하나 버튼(ANC OFF 글자) 둘 다 지원하게 짜드릴게요. 핵심은:

- noiseSource : 음본 소음 재생
- ancSource : 같은 클립을 재생되되, OnAudioFilterRead에서 샘플을 -1배 해서 반대위상으로 만드는 오디오
- ANC ON → ancSource 5. 같이 Play (또는 볼륨 1)
- ANC OFF → ancSource Stop (음본은 계속)

0) 선 구성(필수)

1. Noise 오브젝트에 AudioSource 추가 → 소음 클립 넣기, Loop 체크
2. ANC 오브젝트 하나 더 만들고 AudioSource 추가 → 같은 소음 클립, Loop 체크
3. ANC 오브젝트의 AudioSource에는 아래 PhaseInverter 스크립트를 붙여서 반대위상 되게 함

1) 반대위상 필터 스크립트 (ANC 오브젝트에 붙임)

각 스타트 anc on/off 버튼 슬라이더는 1초마다한번씩, 로그가 찍히도록 해줘

함 "버튼 클릭 순간 로그" + "슬라이더 값은 1초마다 1번 로그(변했을 때만/매번 중 선택)"을 찍게 만들면 됩니다. (크린샷처럼 START/STOP/ANC(도글) + RPM 슬라이더 구성 기준)

합 스크립트 (1초마다 슬라이더 로그 + 각 버튼 로그)

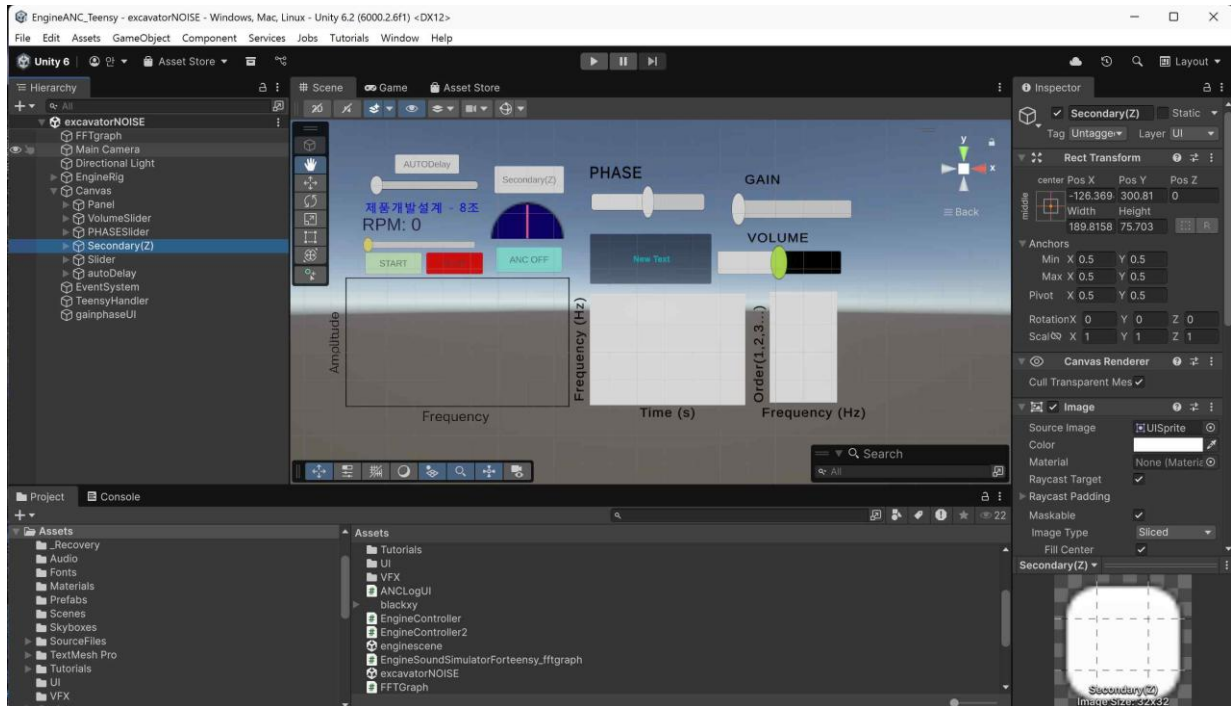
ioLoggerController.cs

```
vp
ng UnityEngine;
ng UnityEngine.UI;

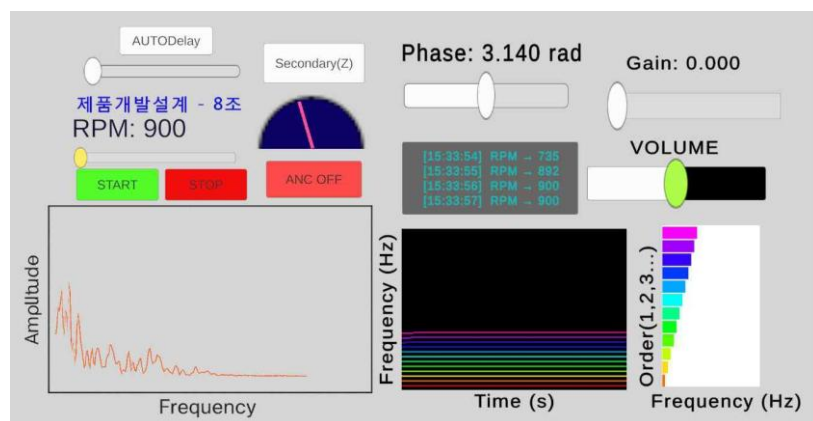
lic class UIAudioLoggerController : MonoBehaviour
{
}
```

[그림18] 캡처한 요소 하나씩 입력 및 동작 코드 요 청

결과적으로, 앱 개발 및 UI/UX 경험이 전혀 없는 상태에서도, “화면 구성 → UI 요소 생성 및 배치 → 캡처 기반 피드백 → 기능 요구 정의 → 코드 생성 및 적용 → 동작 검증”의 반복 과정을 시의 도움만으로 수행함으로써, 실시간 제어가 가능한 시각화 인터페이스를 구현할 수 있었다.



[그림19] Unity – Teensy 통신 능동소음제어시스템 구현



[그림20] Unity – Teensy 통신 시스템 실행화면

6. 모바일 환경을 고려한 앱 확장 구상

최종적으로는 컴퓨터 환경에 국한되지 않고, 모바일 기기에서도 소음을 재생·제어할 수 있는 방향으로 확장을 고려하였다.

이에 대해 산지니 AI를 활용하여, Unity 기반 제어 시스템을 모바일 환경으로 확장하는 개념 설명, 휴대폰에서 소음제어에 대한 이해를 통해, 산지니의 안내만을 기반으로 구현 방향을 구체화할 수 있었다.

유니티로 만든 파일을 휴대폰에서 실행시키고 싶은데, 앱처럼 다운받으려면 어떻게 해야 하는지 하나하나 자세하게 알려줘 초등학교도 이해할수있도록

유니티로 만든 멋진 게임이나 앱을 내 휴대폰에 넣어서 친구들에게 자랑하고 싶군요! 정말 멋진 생각이예요.

컴퓨터에 있는 유니티 파일을 휴대폰에서 실행되는 '앱'으로 만드는 과정을 '**빌드(Build)**'라고 해요. 마치 레고로 만든 성을 단단하게 본드로 붙여서 장난감 상자에 넣는 것과 비슷하죠.

가장 많이 쓰는 '**안드로이드폰(갤럭시 등)**'을 기준으로, 초등학교도 따라 할 수 있게 아주 쉽게 설명해 줄게요.

1단계: 준비물 챙기기 (유니티 설정)

먼저 유니티에게 "이건 컴퓨터용이 아니라 휴대폰용이야!"라고 알려줘야 해요.

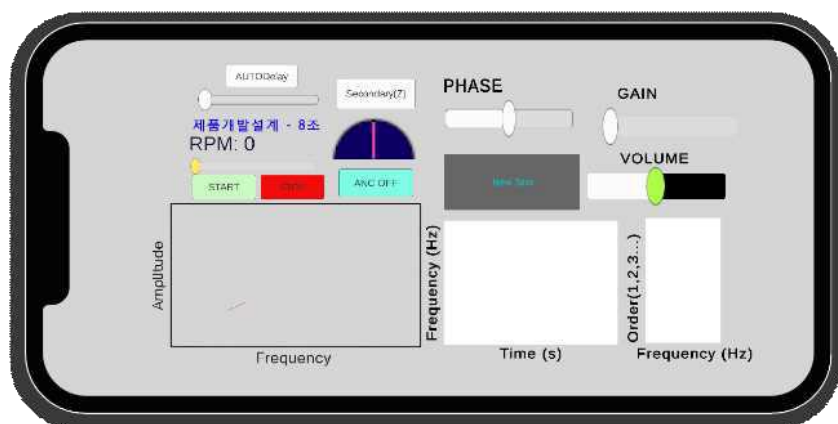
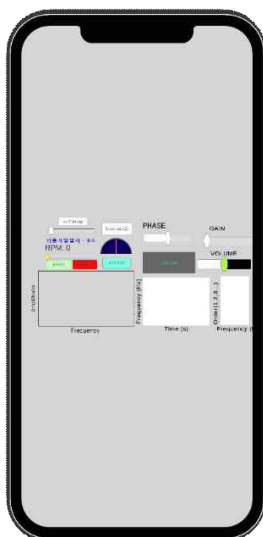
1. **유니티 허브(Unity Hub)**를 켜세요.
2. 왼쪽 메뉴에서 **[Installs]** (설치)를 누르세요.
3. 지금 쓰고 있는 유니티 버전 옆에 있는 **톱니바퀴 모양**을 누르고 **[Add modules]** (모듈 추가)를 선택하세요.
4. 목록에서 ****[Android Build Support]****를 찾아서 체크하세요. (그 아래에 있는 **OpenJDK**, **Android SDK & NDK Tools**도 꼭 같이 체크되어 있어야 해요!)
5. **[Install]** (설치) 버튼을 누르고 설치가 끝날 때까지 기다리세요.

2단계: 휴대폰 개발자 모드 켜기

휴대폰에게 "나 이제 앱 만들 거니까 문 좀 열어줘!"라고 허락을 받아야 해요.

1. 휴대폰 **[설정]** 앱을 켜세요.

[그림21] 앱 개발 관련 프롬프트 작성



[그림22] 모바일 어플리케이션 전환

전문적인 앱 개발 경험 및 지식 없이도, 오직 AI의 지원을 바탕으로 소음제어 시스템을 모바일 환경까지 확장하는 개념적 설계가 가능하였으며, 소음 제어 시스템을 보다 직관적으로 체험할 수 있는 형태로의 확장 가능성을 확보하였다.

○ 정량적 성과

실험 과목 또는 개인 연구의 경우, 수기로 기록된 데이터를 엑셀로 전사하고 그래프를 작성한 뒤 결과를 정리하는 데 기존에는 실험 1회당 약 50-70분이 소요되었다. 산지니 AI를 활용한 이후에는 실험 노트 촬영 후 데이터 판독, 표 정리, 그래프 생성까지의 과정이 통합되어 약 20-30분 내에 완료되었다. 이에 따라 실험 1회 기준 약 30-40분의 작업 시간이 단축되었으며, 주 2-3회 실험이 반복되는 주차에서는 주당 약 1-2시간 수준의 시간 절감 효과가 발생하였다. 또한 데이터 입력 과정에서 발생하던 오타·누락으로 인한 재입력 및 그래프 재작성 빈도가 크게 감소하였다.

ANC 캡스톤 프로젝트에서는 문헌 조사, 하드웨어 구성 검증, 코드 구조 설계 과정에서 반복적인 시행착오가 줄어들었다. 특히 논문·데이터시트 탐색 및 시스템 구조 파악에 소요되던 초기 준비 시간이 기존 대비 약 절반 수준으로 단축되었으며, 핀 연결 오류나 코드 구조 문제로 인한 재작업 시간이 현저히 감소하여 전체 시스템 구현 단계에 더 빠르게 도달할 수 있었다.

○ 정성적 성과

반복적인 입력·정리 작업에서 해방되면서, 실험 결과의 물리적 의미와 오차 원인 분석에 더 많은 시간을 투입할 수 있게 되었다. ANC 프로젝트에서도 전기·전자·제어·소프트웨어가 결합된 영역에 대해 “접근이 불가능한 영역”이라는 인식에서 벗어나, 문제를 단계적으로 분해하여 해결할 수 있다는 인식 변화가 나타났다. 또한 Unity 기반 시각화 인터페이스를 통해 소음 제어 상태를 실시간으로 확인할 수 있게 되면서, 연구 결과의 전달력과 프로젝트 결과물에 대한 체감 완성도가 뚜렷하게 향상되었다.

○ 결론

본 사례에서의 AI 활용 방식은 특정 주제에 한정된 기법이 아니라, 연구 분야·프로젝트 성격과 무관하게 적용 가능한 범용 연구 수행 프로세스이다. 전체 연구 흐름을 문헌 조사 → 방향성 설정 → 하드웨어 구성 → 소프트웨어 구현 → 결과 분석 및 확장의 5단계로 구조화함으로써, 복잡한 공학 연구 및 시스템 설계를 더 빠르고 체계적으로 수행할 수 있다.

또한 실제 적용 과정에서 확인했을 때는, PDF 데이터시트/기술문서 기반의 해석과 질의 응답은 ChatGPT보다 Gemini가 더 잘 처리되는 경우가 있어, 상황에 따라 도구를 병행·선택하는 전략이 유효하다.

마지막으로 AI의 출력 품질은 “요청문(프롬프트)”의 구체성에 크게 좌우된다. 예를 들어 AI에게 “아기도 따라할 수 있도록, 초등학생이 이해하고 실행할 수 있도록 하나하나 알려줘” 라고 명시하면, AI가 과정을 한 단계씩(한 스텝씩) 쪼개서 설명해 주기 때문에, 연구 초심자도 절차를 그대로 따라가며 실험·설계·분석을 수행할 수 있다.

3. 기대효과

본 사례를 통해 제시한 AI 기반 연구·설계 수행 방식은, 기존 학부 연구 환경에서 반복적으로 발생하던 비효율과 진입 장벽을 실질적으로 완화할 수 있는 효과를 가진다. 특히 실험 데이터 처리, 복합 시스템 설계, 비전공 영역의 소프트웨어 구현 등 학부생이 단독으로 수행하기 어려웠던 영역에서 다음과 같은 구체적인 기대효과를 확인할 수 있다.

① 실험 데이터 처리 및 분석 과정의 효율성 향상

수기 기록 중심의 실험 데이터가 AI를 통해 즉시 정형 데이터로 변환되고, 표·그래프·기초 통계 분석까지 연속적으로 처리됨으로써, 반복적인 입력 작업에 소요되던 시간을 대폭 절감할 수 있다. 또한 산지니 AI의 MCP기능은 다른 AI모델보다 더 효율적인 워크플로우를 제공한다. 이는 입력 오류 감소와 분석 결과의 신뢰도 향상으로 이어져 실험 결과의 완성도를 높이는 효과를 가진다.

② 복합 시스템 연구에 대한 접근성 향상

ANC와 같이 기계, 전기·전자, 제어, 소프트웨어가 동시에 요구되는 고난도 시스템의 경우, 학부 수준에서는 문제 접근 자체가 큰 부담으로 작용한다. 본 사례에서 제시한 AI 활용 방식은 문헌 조사, 하드웨어 구성, 코드 구조 설계, 통신 연동까지의 과정을 단계적으로 분해하여 제시함으로써, 전공 외 영역에 대한 심리적·기술적 장벽을 효과적으로 낮춘다.

③ 비전공 영역에 대한 학습 및 구현 가능성 확대

UI/UX 설계, 앱 개발, 시리얼 통신, 소프트웨어 연동과 같은 영역은 기존 기계공학 교과과정에서 거의 다루지 않는 분야이다. 그러나 산지니AI의 단계적 안내를 활용할 경우, 관련 지식과 경험이 전무한 상태에서도 구현 방향을 설정하고 실제 시스템에 적용할 수 있음을 확인하였다. 이는 학부생이 자신의 전공에 국한되지 않고 융합적 문제 해결 역량을 확장할 수 있는 기반을 제공한다.

④ 학부 연구 수행 방식 자체의 구조적 개선이다.

“문헌 조사 → 방향성 설정 → 하드웨어 구성 → 소프트웨어 구현 → 결과 분석 및 확장”의 AI 기반 프로세스는 다양한 공학 연구 및 설계 과제에 동일하게 적용 가능하다. 이를 통해 학부 연구는 시행착오 중심의 반복 작업에서 벗어나, 보다 체계적이고 재현 가능한 연구 수행 구조로 전환될 수 있다.

종합적으로, 산지니 AI는 단순 보조 도구가 아닌 연구 전 과정의 구조를 개선하는 실질적 수단으로 활용할 수 있으며, 학부 수준에서도 복잡한 공학 문제를 효율적으로 수행할 수 있는 연구 수행 패러다임을 제시한다.