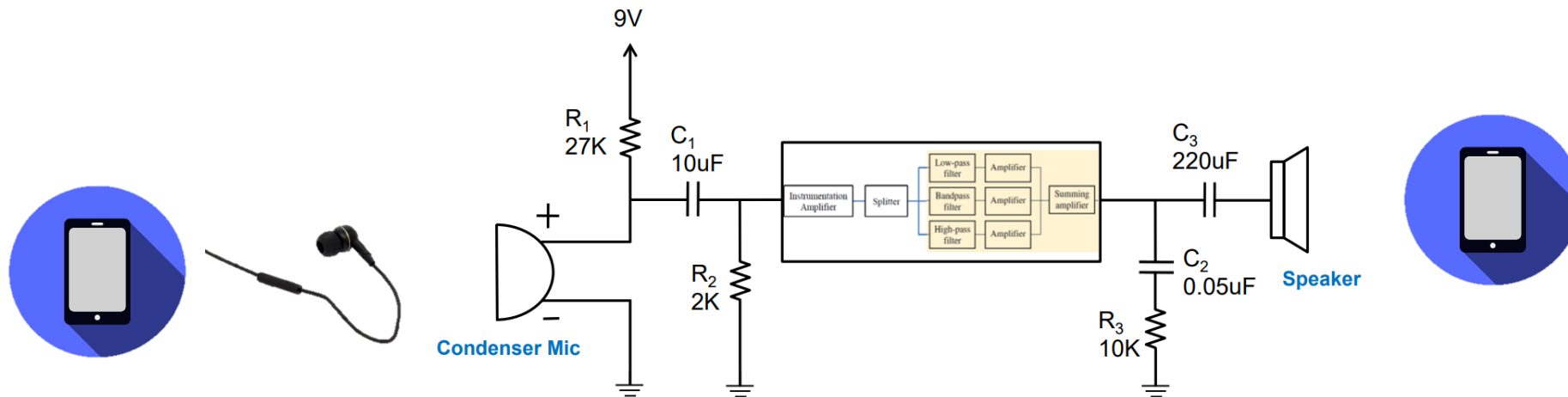


전자공학실험 Term Project

Term project

- Sound equalizer

- ✓ Condenser mic.에 음성(speech)/음악(music)을 핸드폰, 노트북 등으로 재생
- ✓ Speaker에 핸드폰, 노트북 등 녹음이 가능한 기기를 이용하여 녹음
 - m4a로 저장되는 경우 <https://convertio.co/kr/m4a-wav/> 또는 <https://online-audio-converter.com/ko/> 등을 이용하여 wav 확장자명으로 저장
 - 최대한 조용한 환경에서 재생/녹음 진행
 - 가급적 재생/녹음 시작과 중지 모두 동시에 진행
 - 재생 음원이 녹음 장치에 녹음되지 않도록 주의








녹음 환경 예시

Term project

- 실험 분석

- ✓ 아래 github 주소에서 음원 및 스크립트 다운로드
 - https://github.com/mystlee/21_elec_exp
- ✓ 실험 방법은 개인 PC에서 실행하는 방법과 Google Colab을 이용하여 온라인에서 진행하는 방법이 있음
 - Colab에서 진행하는 방법은 PC에 설치를 요구하는 것이 없음
 - 자세한 내용은 github의 readme.md 참조
- ✓ 음원 샘플은 총 5개이며, samples폴더에 존재하고 파일 이름에 샘플 번호가 아래와 같이 표기
 - Ex) mixed_1.wav (1번 음원 샘플)




 mixed_1.wav	2023-11-08 오전 2:11	WAV 파일	409KB
 mixed_2.wav	2023-11-08 오전 2:28	WAV 파일	320KB
 mixed_3.wav	2023-11-08 오전 2:33	WAV 파일	472KB
 mixed_4.wav	2023-11-08 오전 2:37	WAV 파일	472KB
 mixed_5.wav	2023-11-08 오전 2:39	WAV 파일	378KB

Term project

- 실험 분석

- ✓ 녹음된 오디오 파일을 아래와 같은 경로에 저장

- 설계한 회로에서 재생되는 음원을 wav 파일로 변환한 뒤, recorded 폴더에 mixed_{sample_idx}_{filter}.wav 로 저장
 - {sample_idx}에는 샘플 음원의 번호를, {filter}에는 실험하는 filter의 종류를 삽입
 - Ex) mixed_1.wav 파일 (음원 샘플 1번)으로 lowpass를 실험하는 경우 mixed_1_lowpass.wav

이름	수정된 날짜	유형	크기
 mixed_1_bandpass.wav	2023-11-08 오전 3:04	WAV 파일	407KB
 mixed_1_highpass.wav	2023-11-08 오전 3:06	WAV 파일	407KB
 mixed_1_lowpass.wav	2023-11-08 오전 3:05	WAV 파일	407KB

Term project

- 실험 분석

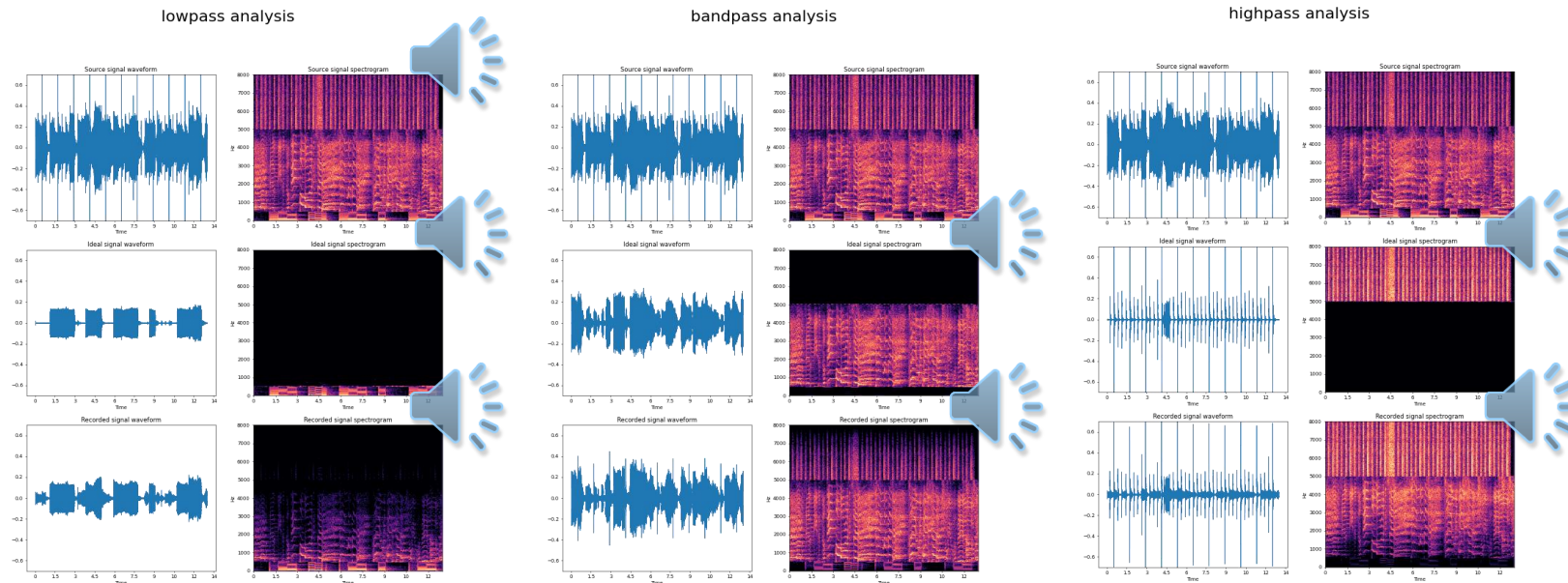
- ✓ main.py 또는 main.ipynb에서 'sample_idx' 와 'filter_mode' 변수만을 수정
 - 'sample_idx'는 음원 샘플의 index를 나타내는 것으로 1~5사이의 정수 값을 입력
 - Lowpass를 실험하는 경우 'filter_mode'를 'lowpass', highpass를 실험하는 경우 'highpass', bandpass를 실험하는 경우 'bandpass'로 설정하여 실험

```
✓ [4] # MODIFY THE SAMPLE AND FILTER
      sample_idx = 1 # 1 ~ 5
      filter_mode = 'lowpass' # lowpass, highpass, or bandpass
```

Term project

- 실험 분석

- ✓ 입력 신호(source signal), 이상적인 필터 통과 신호(ideal signal), designed 필터 통과 신호에 대한 waveform 및 spectrogram 비교
 - Spectrogram은 시간-주파수에 대한 표현으로 x축은 시간, y축은 주파수를 표기
 - 스크립트 실행 시, 우측 그림과 같은 그림 파일 생성
 - 제일 상단은 source signal, 가운데는 ideal signal, 제일 하단은 designed signal을 나타냄



Term project

- 실험 분석

- ✓ 입력 신호(source signal), 이상적인 필터 통과 신호(ideal signal), designed 필터 통과 신호에 대한 waveform 및 spectrogram 비교
 - Spectrogram은 시간-주파수에 대한 표현으로 x축은 시간, y축은 주파수를 표기
- ✓ Signal-invariant signal-to-noise ratio (SI-SNR) 객관적 지표를 사용하여, ideal 신호와 designed 신호와의 유사도 평가
 - 신호와 처리 과정에서 발생한 왜곡된 신호에 대한 비율을 나타내는 객관적 지표

$$SDR := 10 \log_{10} \left(\frac{\|\mathbf{s}_{target}\|^2}{\|\mathbf{e}_{noise}\|^2} \right), \text{ where, } \mathbf{s}_{target} = \frac{\langle \hat{\mathbf{S}}, \mathbf{s} \rangle}{\|\mathbf{s}\|^2} \mathbf{s}, \quad \mathbf{e}_{noise} = \hat{\mathbf{S}} - \mathbf{s}_{target}$$

\mathbf{s}_{target} : target signal (ideal 신호), \mathbf{e}_{noise} : noise signal (filtering 이후 residual 신호)

- Designed 신호가 ideal 신호와 유사할 수록 높은 score를 나타냄

Thank you
Q&A

