

코드 다운로드 : https://github.com/MinjaeKim7280/Picoscope_data_acquisition

1. 소갯말

이 소프트웨어를 사용하면 채널, 전압 범위, 샘플링 속도 및 지속 시간 설정을 사용자 정의하여 PicoScope 5000 시리즈에서 데이터를 수집할 수 있습니다. 수집된 데이터는 손쉽게 분석할 수 있도록 NumPy(.npy) 형식으로 저장됩니다.

2. 하드웨어 연결

- 1) PicoScope 장치에 전원 케이블을 연결합니다.
- 2) 제공된 USB 케이블을 사용하여 PicoScope 장치와 PC를 연결합니다. (Figure 1 참조)



Figure 1 하드웨어 연결

- 3) PicoScope의 전면부 채널 포트에 프로브(Figure 2)를 연결합니다.



Figure 2 오실로스코프 프로브

- 4) 프로브 끝에 측정하려는 전극을 연결합니다.

3. 소프트웨어 실행

※ 소프트웨어 실행 전 충분한 하드 드라이브 용량을 확보하십시오.

3.1 실행파일

- 1) 프로그램 실행 전 <https://www.picotech.com/downloads>에 접속하여 사용하는 picoscope의 모델 (5444D MSO)을 선택 후, 사용하는 PC의 운영체제(32bit or 64bit)에 맞춰 PicoSDK를 다운로드합니다. (Figure 3 참조)

Product series	PicoScope 5000 Series	PicoScope 5444D MSO	Software				
PicoScope 2000 Series >	PicoScope 5242D MSO >	Software >	Resource	Architecture	Release	Size	Released
PicoScope 3000D Series >	PicoScope 5243D >	Manuals & brochures >	Windows:				
PicoScope 3000E Series >	PicoScope 5243D MSO >		PicoScope 7.1.29 (64-bit)	x64	Stable *Recommended	235 MB	Jun 13 2024
PicoScope 4000 Series >	PicoScope 5244D >		Picoscope 7.1.32 Early Access	x64	Early Access	235 MB	Jul 02 2024
PicoScope 5000 Series >	PicoScope 5244D MSO >		PicoLog 6.2.9		Stable	103 MB	Feb 21 2024
PicoScope 6000 Series >	PicoScope 5442D >		PicoSDK 10.7.26 (32-bit)	x86	Stable	117 MB	Mar 08 2024
PicoScope 9000 Series >	PicoScope 5442D MSO >		PicoSDK 10.7.26 (64-bit)	x64	Stable	121 MB	Mar 08 2024
PicoLog Data Loggers >	PicoScope 5443D >						
PicoSource >	PicoScope 5443D MSO >						
PicoVNA >	PicoScope 5444D >						
Discontinued products >	PicoScope 5444D MSO >						

Figure 3 PicoSDK download

- 2) 다운로드된 PicoSDK설치파일을 실행하여 PicoSDK를 설치합니다. (기본값으로 설치)
- 3) 설치 후, PicoProbe.exe를 실행합니다. (실행 오류 시 [3.2 cmd 실행] 참고)

- 로딩이 오래 걸릴 수 있습니다.

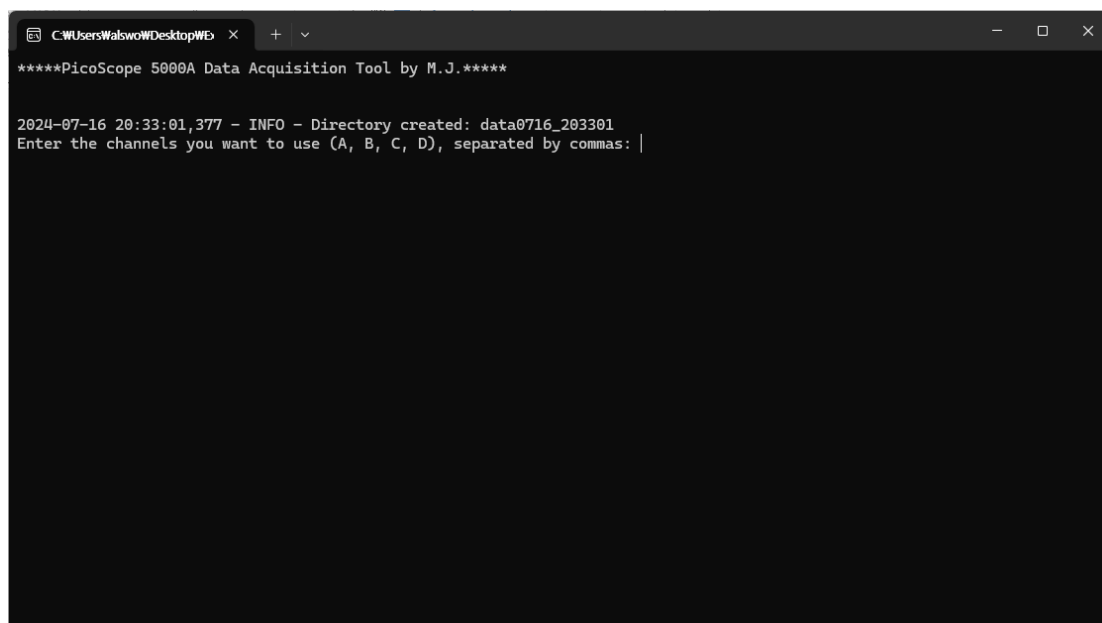


Figure 4 PicoProbe.exe

- 4) 실행 후, [4. 소프트웨어 초기 설정]으로 이동하여 소프트웨어를 사용합니다.

3.2 cmd 실행

※ 실행 파일 오류 시 또는 코드 수정 시 아래 방법을 통해 실행이 가능합니다.

1) 실행 전 모듈 설치 및 버전 확인 (개발 버전) (anaconda 가상 환경 구축 추천)

Python 3.7.1

Numpy 1.21.5

Pandas 1.3.5

Picosdk 1.1 ([3.1 실행파일 실행] PicoSDK 다운로드 참고)

2) 윈도우+R을 누르고 실행창이 뜨면 cmd를 입력하여 명령 프롬프트를 실행합니다.

(anaconda의 경우 anaconda prompt로 이동)

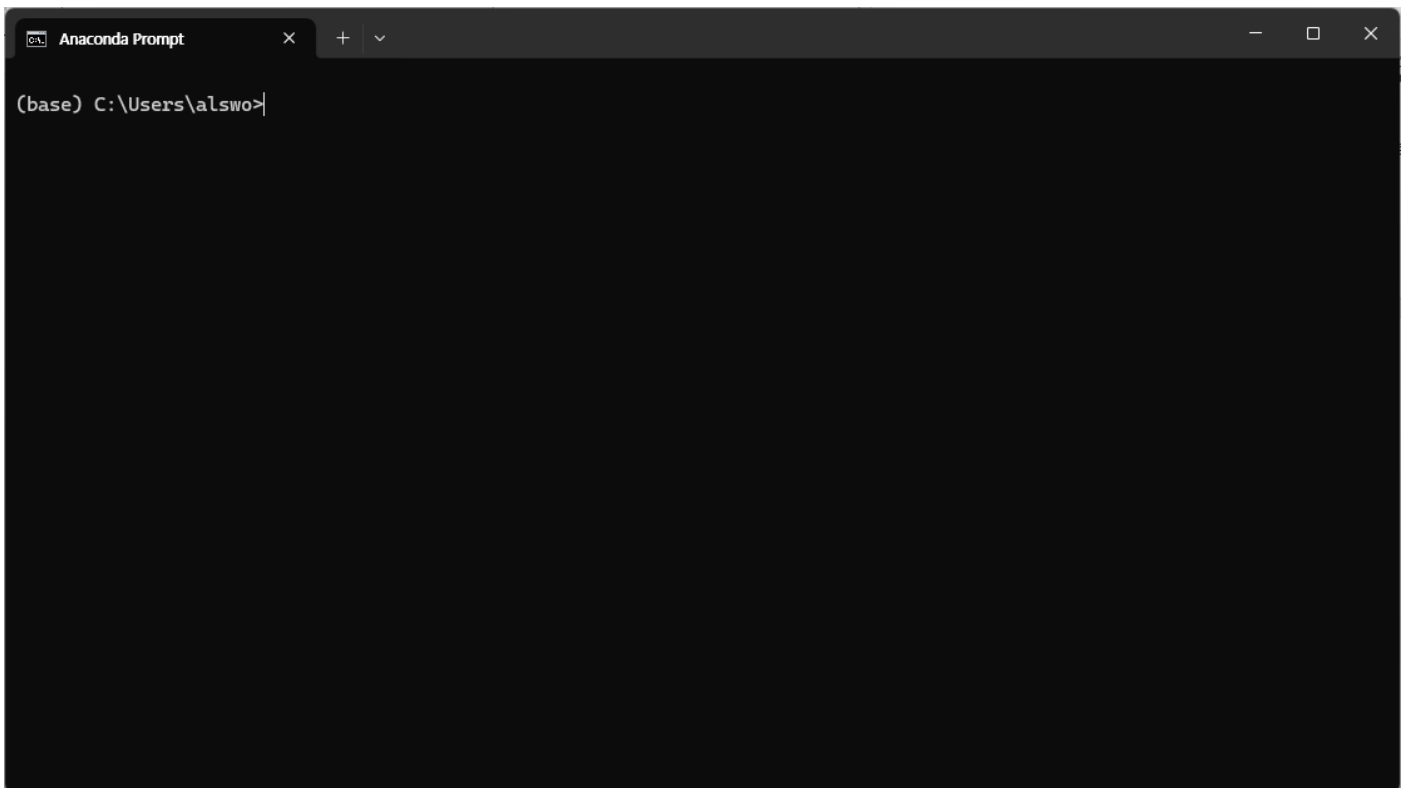


Figure 5 Anaconda prompt

3) PicoProbe.py 파일이 있는 폴더로 명령 프롬프트의 경로를 이동합니다. (예: cd example)

※ 파일 탐색기에서 해당 폴더를 클릭하여 복사 + 붙여넣기하면 간단합니다. (Figure 6)

명령어 : cd C:\Users\alswo\Desktop\Example

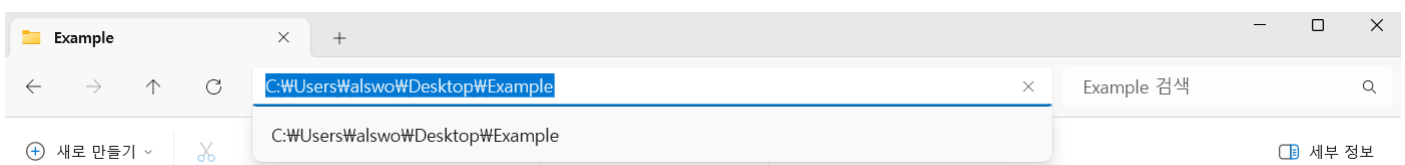


Figure 6 Directory path

cd Desktop -> cd Example 순으로 명령어를 입력하여도 동일합니다.

5) python PicoProbe.py를 입력하여 파일을 실행합니다.

```
(picoscope) C:\Users\alswo\Desktop\Example>python PicoProbe.py
*****PicoScope 5000A Data Acquisition Tool by M.J.*****

2024-07-16 20:52:33,850 - INFO - Directory created: data0716_205233
Enter the channels you want to use (A, B, C, D), separated by commas: |
```

Figure 7 Python file execution

6) 실행 후, [4. 소프트웨어 초기 설정]으로 이동하여 소프트웨어를 사용합니다.

4. 소프트웨어 초기 설정

PicoProbe.py 실행 시, 실행시간을 기준으로 데이터가 저장될 폴더가 해당 파일이 있는 경로에 자동으로 생성됩니다. 이후 채널 선택 등의 설정 절차가 진행됩니다. (Figure 8 참고)

1) 사용 Channel 선택

사용 가능한 Channel은 A, B, C, D 이며, 프로브를 연결하고 사용을 원하는 채널을 입력합니다. 채널 선택은 쉼표(,)로 구분하며 대소문자는 구분하지 않습니다. (예: A,B 또는 A,c,D)

2) Voltage Range 선택

선택한 채널별로 Voltage range를 설정이 가능합니다. 각 채널별로 아래 옵션 중 하나를 선택하여 입력합니다.

(10mV, 20mV, 50mV, 100mV, 200mV, 500mV, 1V, 2V, 5V, 10V, 20V) 기본값 : 2V

****단위 포함 입력, 대소문자 구분**

3) Sampling Rate 설정 [7. 경고 & 오류] 참고 추천

샘플을 측정하는 주기인 샘플링 속도를 설정합니다. 단위는 ns(나노초)이며, 너무 낮게 설정하면 오버플로우가 발생할 수 있으므로 주의해야 합니다.

****숫자만 입력**

4) Resolution 설정

8 또는 12를 입력합니다. 이 값은 데이터의 해상도를 의미하며, 12비트를 선택하면 용량이 커지는 대신 더 높은 해상도의 데이터를 얻을 수 있습니다. ****숫자만 입력**

5) Duration 설정

프로그램의 실행 시간을 설정합니다. time 또는 manual을 입력합니다.

- Time 입력 시 측정할 시간(초)을 입력합니다. 예를 들어 10초 입력 시 측정은 10초 동안만 진행됩니다. ****숫자만 입력**
- Manual 입력 시 사용자가 직접 중지할 때까지 측정이 진행됩니다. Ctrl+C를 입력하여 측정을 종료할 수 있습니다.

```
2024-07-15 21:03:54,611 - INFO - Directory created: data0715_210354
Enter the channels you want to use (A, B, C, D), separated by commas: A,c,D

Voltage range options: 10mV, 20mV, 50mV, 100mV, 200mV, 500mV, 1V, 2V, 5V, 10V, 20V
Enter voltage range for channel A: 2V

Voltage range options: 10mV, 20mV, 50mV, 100mV, 200mV, 500mV, 1V, 2V, 5V, 10V, 20V
Enter voltage range for channel C: 5V

Voltage range options: 10mV, 20mV, 50mV, 100mV, 200mV, 500mV, 1V, 2V, 5V, 10V, 20V
Enter voltage range for channel D: 2V
Enter sampling rate (in ns): 200
Enter resolution (8 or 12 bit): 12
Enter 'time' for specific duration or 'manual' for manual termination: time
Enter duration in seconds: 10
```

Figure 8 Initial Settings Example

5. Data 측정

[4. 소프트웨어 초기 설정]을 완료하면 즉시 측정이 시작됩니다.

1) Device 연결, 입력값에 따른 측정 설정들의 로그를 출력합니다. (Figure 9)

```
2024-07-15 21:39:10,687 - INFO - Device opened successfully
2024-07-15 21:39:10,687 - INFO - Channels A, C, D set up successfully
2024-07-15 21:39:10,688 - INFO - Buffers set successfully
2024-07-15 21:39:10,806 - INFO - Streaming started with sample interval: 208 ns
```

Figure 9 Initial log

2) 실행 중에는 각 채널에 저장되는 샘플 수가 실시간으로 표시됩니다. 이를 통해 측정 진행 상황을 확인할 수 있습니다.

```
2024-07-15 21:39:12,942 - INFO - Put in Queue 10092544
2024-07-15 21:39:13,197 - INFO - Processing data batch 0
2024-07-15 21:39:13,216 - INFO - Data 0 saved for channel A, Sample #: 10092544
2024-07-15 21:39:13,232 - INFO - Data 0 saved for channel C, Sample #: 10092544
2024-07-15 21:39:13,250 - INFO - Data 0 saved for channel D, Sample #: 10092544
```

Figure 10 Data Saving log

3) 해당 데이터들은 프로그램 실행 시 생성된 폴더(data0715_210354) 안, 각 채널 폴더에 저장됩니다.



Figure 11 Data Files

Figure 12 Channel Folders

4) 종료 시 데이터 측정 시간, 데이터 폴더 위치 등 종료 로그가 출력됩니다.

```
2024-07-15 21:39:20,839 - INFO - Specified duration of 10 seconds reached. Stopping data collection.
2024-07-15 21:39:20,839 - INFO - Data collection stopped.
2024-07-15 21:39:20,849 - INFO - Process time: 10.043707370758057 seconds
2024-07-15 21:39:20,858 - INFO - Save data worker finished
2024-07-15 21:39:20,899 - INFO - Files saved to data0715_210354
2024-07-15 21:39:20,985 - INFO - Device closed
2024-07-15 21:39:20,985 - INFO - Data saved and program terminated
```

Figure 13 Termination log

5) 해당 데이터는 PicoScope에서 측정되는 ADC 데이터를 가공없이 저장하는 것이며, 데이터의 type은 numpy.int16 입니다.

6. 데이터 분석

6.1 데이터 그래프

- 1) Data_Plot.py를 실행하여 데이터 그래프를 확인할 수 있습니다. (해당 파일은 matplotlib, numpy의 설치 필요)
- 2) 데이터의 경로를 입력합니다. 하나의 데이터만 확인하는 경우(Figure 14), 하나의 경로를 입력하고 데이터 간의 연결 지점을 확인(Figure 15) 하려면 쉼표(,)로 구분하여 두 개의 경로를 입력합니다.

※ 연결 그래프 확인 시 연속된 데이터 입력

```
(picoscope) C:\Users\alswo\Desktop\Example>python data_plot.py
Enter the path(s) to your .npz file(s), separated by comma if two files: C:\Users\alswo\Desktop\Example\dist\data0716_215214\channel_a\data_5.npz
```

Figure 14 Single Data plot Path input

```
(picoscope) C:\Users\alswo\Desktop\Example>python data_plot.py
Enter the path(s) to your .npz file(s), separated by comma if two files: C:\Users\alswo\Desktop\Example\dist\data0716_215214\channel_a\data_5.npz, C:\Users\alswo\Desktop\Example\dist\data0716_215214\channel_a\data_6.npz
```

Figure 15 Multiple Data plot Path input

- 3) 데이터를 측정할 전압 범위, sampling rate를 입력합니다. ****숫자만 입력**

```
Enter the voltage range (in Volts): 2
Enter the sampling interval (in ns): 48
```

Figure 16 Voltage range & Sampling Interval input

- 4) 그래프가 출력됩니다.

6.2 데이터 로드

그래프를 제외한 다른 분석을 위해 데이터의 로드가 필요할 경우, np.load(file_path) 를 통해 데이터 로드가 가능합니다.

7. 경고 & 오류

해당 소프트웨어의 경우 오류 핸들링이 완벽하지 않으니 오류 발생이 잦을 수 있습니다.

오류가 발생하면 일반적으로 프로그램 재시작 및 PicoScope 재연결 을 추천합니다.

큐 용량 경고 ※ 빈번하게 발생할 수 있습니다.

- 경고 메시지: "Queue is filling up" 또는 "Queue is critically full"
 - 해결 방법:
 - a) 데이터 저장 속도가 데이터 수집 속도를 따라가지 못하고 있습니다.
 - b) Sampling rate를 낮추거나 선택한 채널 수를 줄여야 합니다.
 - c) 테스트 과정에 해당 오류가 발생하지 않은 sampling rate의 최솟값은 아래와 같습니다.
(1Ch. : 48ns / 2Ch. : 80ns / 3Ch. : 112ns / 4Ch. 160ns)
- ※ 해당 sampling rate는 하드웨어 및 실행 환경에 따라 달라질 수 있습니다.

초기 설정값 입력 오류

- 경고 메시지: "Invalid ... input"
- 해결 방법:
 - a) 각 설정에 대한 지시사항을 숙지하고 다시 입력바랍니다.

1) 장치 연결 오류

- 오류 메시지: "Error opening device" 또는 "Device not found"
- 해결 방법:
 - a) PicoScope가 컴퓨터에 제대로 연결되어 있는지 확인
 - b) USB 케이블을 분리했다가 다시 연결
 - c) 다른 USB 포트를 사용
 - d) PicoScope 드라이버가 올바르게 설치되어 있는지 확인

2) 전원 관련 오류

- 오류 메시지: "Power source issue" 또는 "Insufficient power"
- 해결 방법:
 - a) PicoScope에 충분한 전원이 공급되고 있는지 확인
 - b) 전원 어댑터를 사용 중이라면 제대로 연결되어 있는지 확인

3) 채널 설정 오류

- 오류 메시지: "Error setting up channel"
- 해결 방법:
 - a) 선택한 채널이 PicoScope에서 지원되는지 확인
 - b) 전압 범위 설정이 올바른지 확인
 - c) 프로그램을 재시작하고 채널 설정을 다시 시도

4) 버퍼 설정 오류

- 오류 메시지: "Error setting up buffer"
- 해결 방법:
 - a) 컴퓨터의 가용 메모리를 확인
 - b) 프로그램을 재시작하고 다시 시도
 - c) 버퍼 크기 축소 (코드 수정 필요).

5) 데이터 수집 중 오류

- 오류 메시지: "Error in main_loop" 또는 "Error in data collection"
- 해결 방법:
 - a) PicoScope와의 연결이 안정적인지 확인
 - b) 선택한 Sampling rate가 PicoScope의 성능 범위 내인지 확인
 - c) 프로그램을 재시작하고 다시 시도

6) 데이터 저장 오류

- 오류 메시지: "Error in save_data_worker"
- 해결 방법:
 - a) 출력 폴더에 대한 쓰기 권한이 있는지 확인
 - b) 디스크 공간이 충분한지 확인
 - c) 출력 경로에 잘못된 문자가 포함되어 있지 않은지 확인

7) 프로세스 종료 오류

- 경고 메시지: "Save process did not terminate gracefully"
- 해결 방법:
 - a) 프로그램이 강제 종료되었을 수 있습니다. 데이터 손실 여부를 확인

8. 코드 수정 및 참고

PicoProbe.py, Data_Plot.py는 아래 링크에서 다운로드 가능합니다.

https://github.com/MinjaeKim7280/Picoscope_data_acquisition

코드 수정 및 오류 해결은 코드에 포함된 주석 및 picotech의 사용자 가이드와 programmers guide를 참고하십시오.