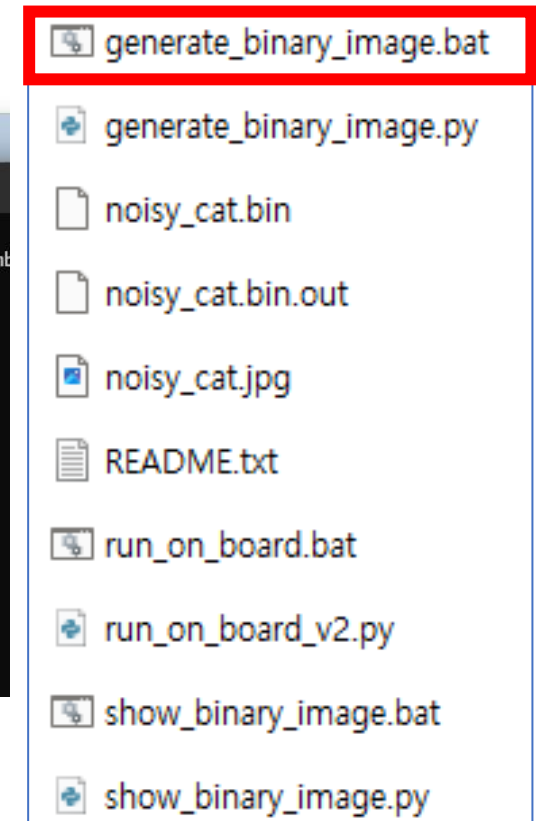
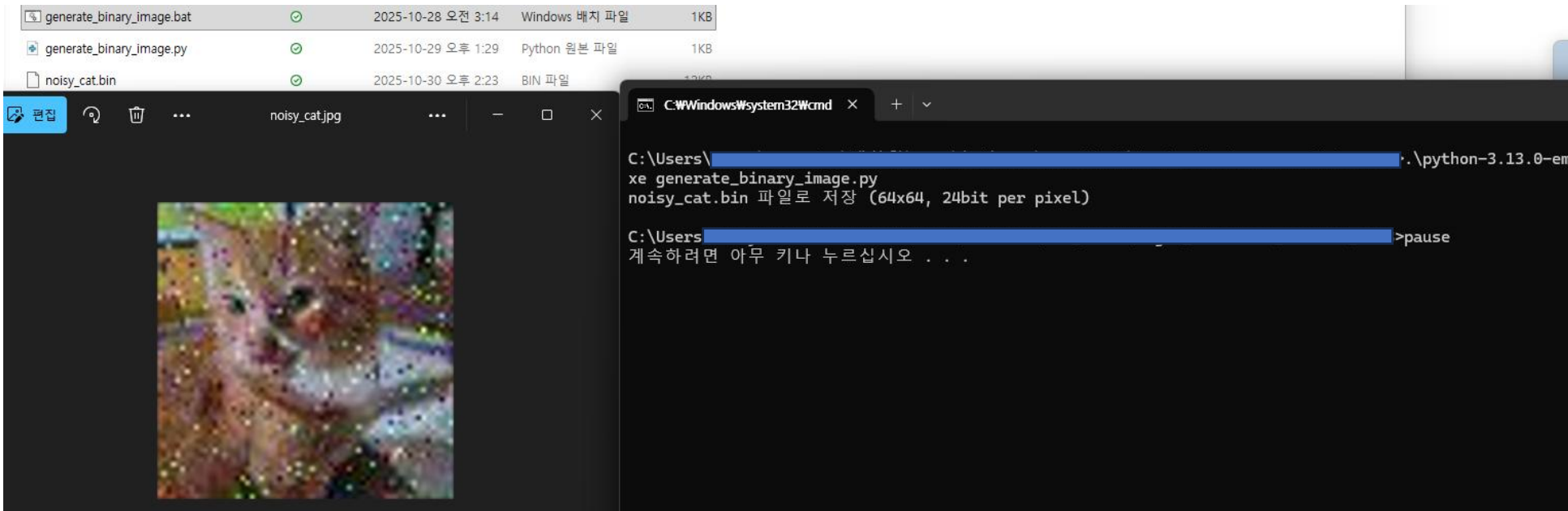


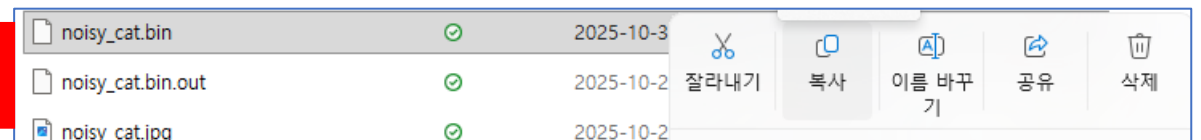
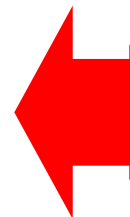
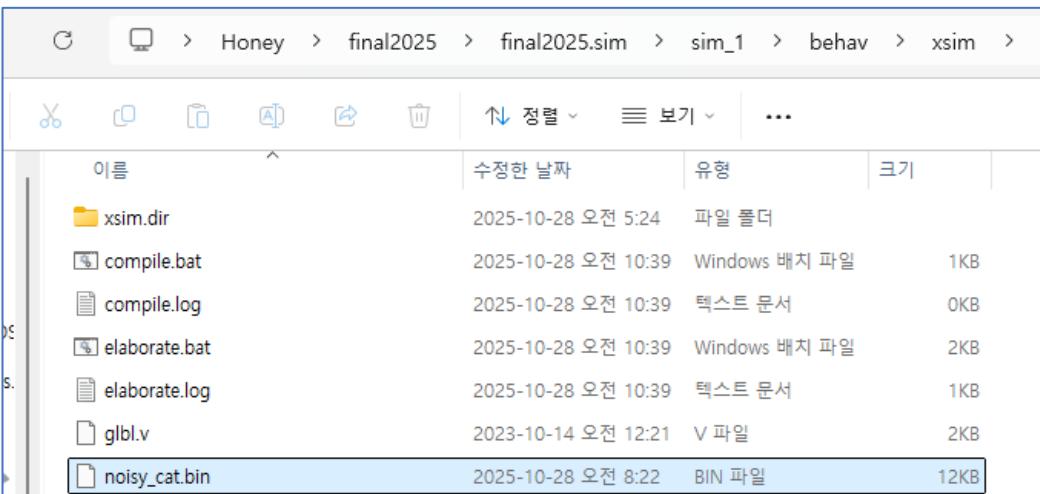
Final project toolkit 사용법

- 1. generate_binary_image.bat 실행
 - noisy_cat.jpg 를 noisy_cat.bin 으로 변환해줌



Final project toolkit 사용법

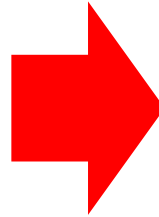
- 2. testbench 에서 실행하기
 - 1) Vivado 프로젝트 생성 후, tb_final.v 파일을 import 한다.
 - 2) (프로젝트 폴더)\(프로젝트명).sim\sim_1\behave\xsim 폴더에 들어가기
 - 3) 폴더 내에 1. 에서 생성한 noisy_cat.bin 파일을 넣기
 - 4) 시뮬레이션 실행
- 참고 : Vivado 내에서 시뮬레이션을 한번도 실행하지 않았을 경우, xsim 폴더가 없을 수 있음, 이 경우 시뮬레이션을 한번 돌린 후 종료하면 정상적으로 생성된다.



Final project toolkit 사용법

- 2. testbench 에서 실행하기
 - 4) (프로젝트 폴더)\(프로젝트명).sim\sim_1\behave\xsim
폴더에서 noisy_cat.bin.out 을 final project toolkit 폴더로 이동
 - 5) show_binary_image 실행하여 결과 이미지 확인

| | | | |
|-------------------|---------------------|---------------|------|
| xsim.dir | 2025-10-29 오후 12:35 | 파일 폴더 | |
| compile.bat | 2025-10-29 오후 12:36 | Windows 배치 파일 | 1KB |
| compile.log | 2025-10-29 오후 12:36 | 텍스트 문서 | 0KB |
| elaborate.bat | 2025-10-29 오후 12:36 | Windows 배치 파일 | 2KB |
| elaborate.log | 2025-10-29 오후 12:36 | 텍스트 문서 | 1KB |
| glbl.v | 2023-10-14 오전 12:21 | V 파일 | 2KB |
| noisy_cat.bin | 2025-10-28 오전 8:22 | BIN 파일 | 12KB |
| noisy_cat.bin.out | 2025-10-29 오후 12:36 | OUT 파일 | 12KB |



| | | | | |
|---------------------------|---|---------------------|---------------|------|
| generate_binary_image.bat | ✓ | 2025-10-28 오전 3:14 | Windows 배치 파일 | 1KB |
| generate_binary_image.py | ✓ | 2025-10-29 오후 1:29 | Python 원본 파일 | 1KB |
| noisy_cat.bin | ✓ | 2025-10-30 오후 2:23 | BIN 파일 | 12KB |
| noisy_cat.bin.out | ✓ | 2025-10-29 오후 2:16 | OUT 파일 | 12KB |
| noisy_cat.jpg | ✓ | 2025-10-28 오전 8:21 | JPG 파일 | 3KB |
| README.txt | ✓ | 2025-04-11 오후 12:35 | 텍스트 문서 | 1KB |
| run_on_board.bat | ✓ | 2025-10-28 오전 9:58 | Windows 배치 파일 | 1KB |
| run_on_board_v2.py | ✓ | 2025-10-29 오후 2:14 | Python 원본 파일 | 7KB |
| show_binary_image.bat | ✓ | 2025-10-28 오전 3:21 | Windows 배치 파일 | 1KB |

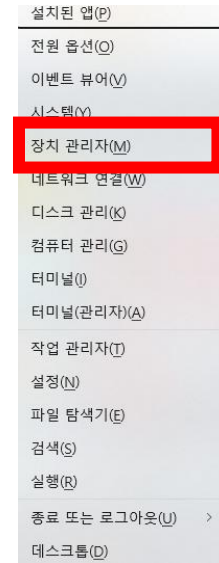


Final project toolkit 사용법

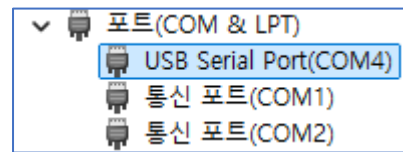
• 3. FPGA에서 결과 확인하기

- 1) Simulation에서 정상 동작이 확인된 경우, generate bitstream 후 보드에 업로드한다.

- 2) win+x 를 눌러 장치관리자를 실행한다.



- 3) 포트를 확인하여 USB Serial Port가 COM포트 몇번에 연결되어있는지 확인한다.



- 4) run_on_board.bat 파일을 우클릭하여 메모장으로 편집을 선택한 후, --port 부분을 앞서 확인한 포트번호에 맞게 변경한다.

```
.Wpython-3.13.0-embed-amd64Wpython.exe run_on_board_v2.py --port COM4 --baud 115200 --width 64 --height 64 --chunk 512
pause
```

Final project toolkit 사용법

- 3. FPGA에서 결과 확인하기
 - 5) run_on_board.bat 파일을 실행하면 FPGA로 UART를 통해 이미지가 전송됨.
 - 6) 이후 show_binary_image.bat 를 실행하면 결과 image 확인 가능

