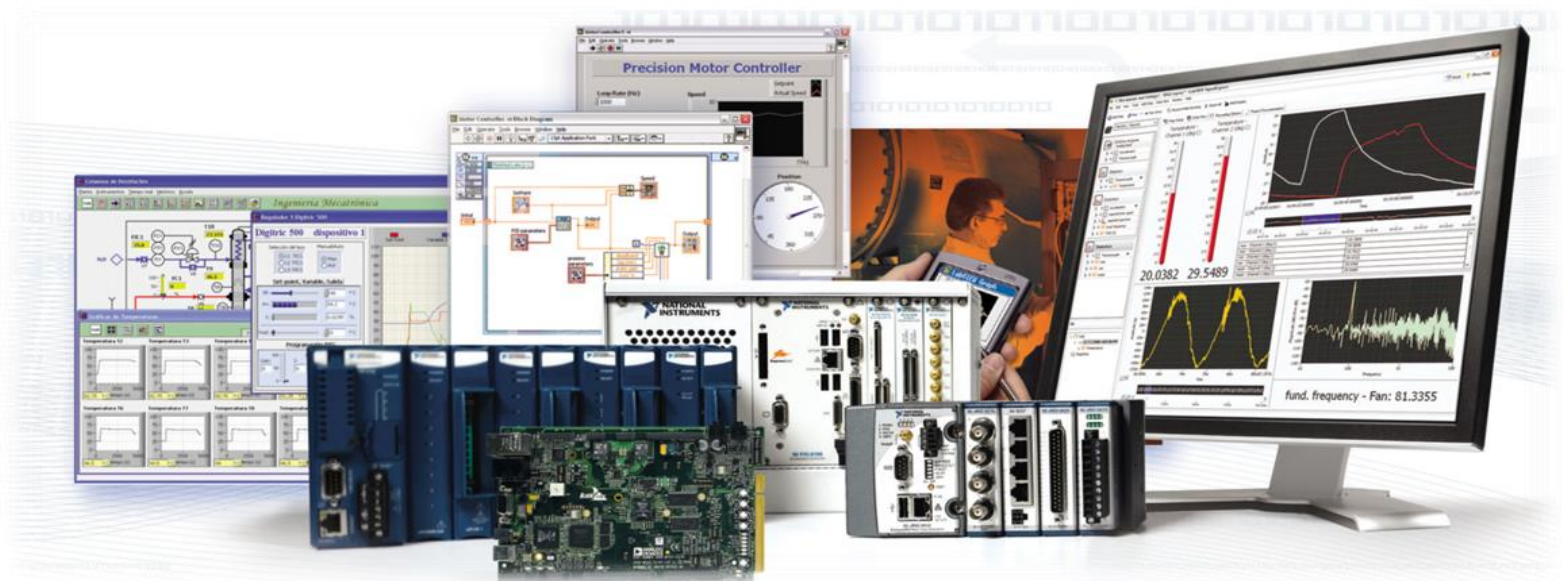


# LabVIEW

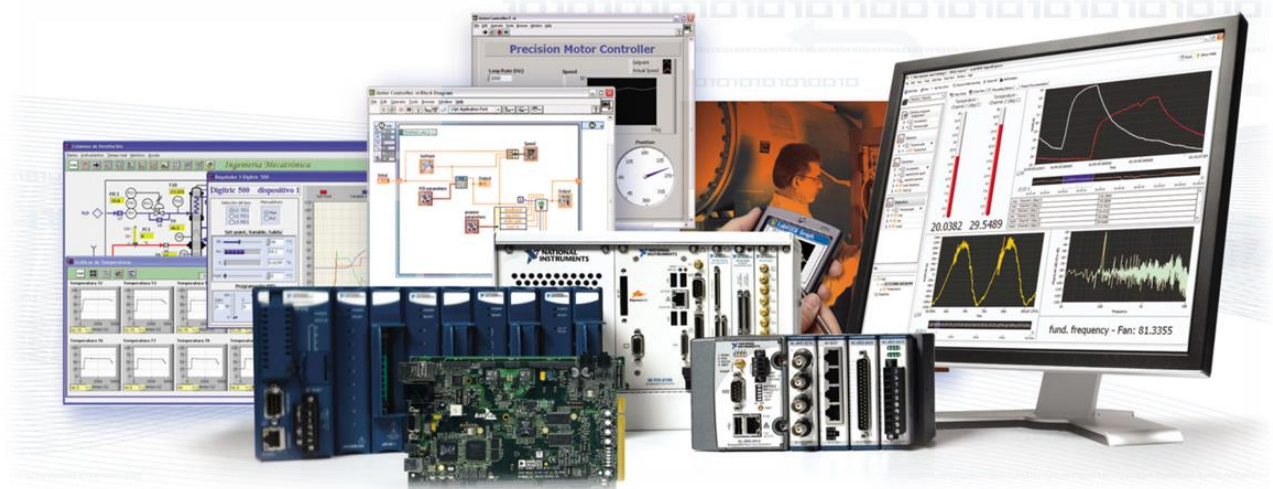
## LabVIEW의 정석 기본편



INFINITYBOOKS

인 피 니 티 북 스

## 6. 파일 입출력





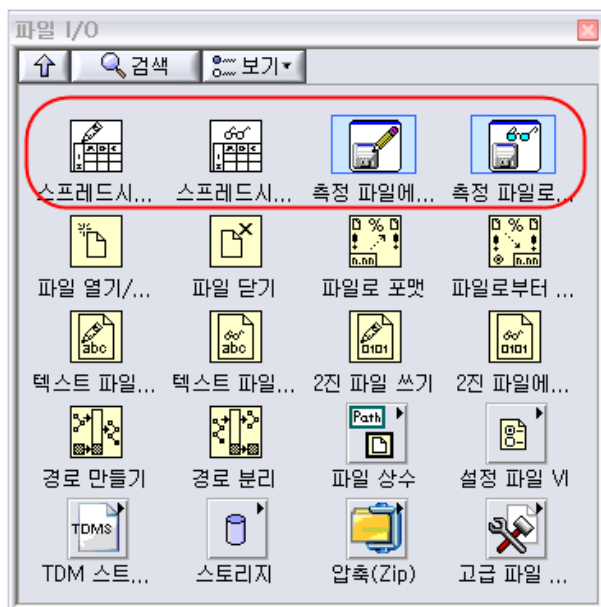
## 파일 포맷

- ❖ 아스키 : 메모장 등에서 데이터 확인 가능
- ❖ 바이너리 : 속도나 용량이 효율적임
- ❖ **LVM : LabVIEW**용 아스키
- ❖ **TDMS : LabVIEW**용 바이너리

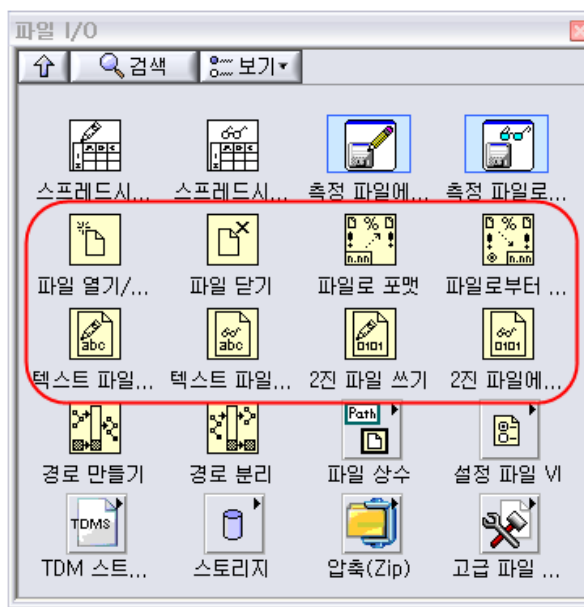


## 상위/하위레벨 노드들

- ❖ 파일 입출력 프로그램을 하는 방법은 상위 레벨 노드를 이용하여 프로그램하거나 하위레벨 노드를 이용하여 프로그램하는 방법이 있다.
- ❖ 전자는 프로그램하기 쉬운 장점이 있으나 사용자 정의가 불가능하다.



상위레벨

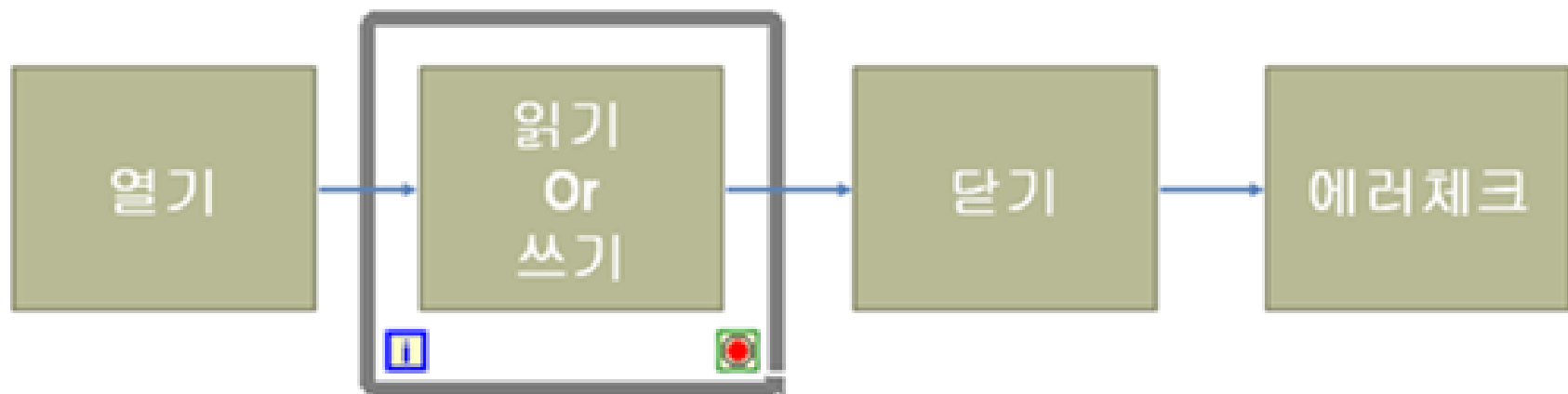


하위레벨



## 상위/하위레벨 노드들

- ❖ 하위레벨 프로그램 순서는 그림과 같다.
- ❖ 쓰거나 읽는 동작의 반복이 필요하면 반복할 코드만 **While** 루프 안에 위치시킨다.
- ❖ '열기'나 '닫기'는 **While** 루프에 넣어 파일을 열고 닫고하는 작업을 반복할 필요가 없다.



파일 열기/생성/대체



텍스트 파일 쓰기

or



텍스트 파일에서 읽기



파일 닫기

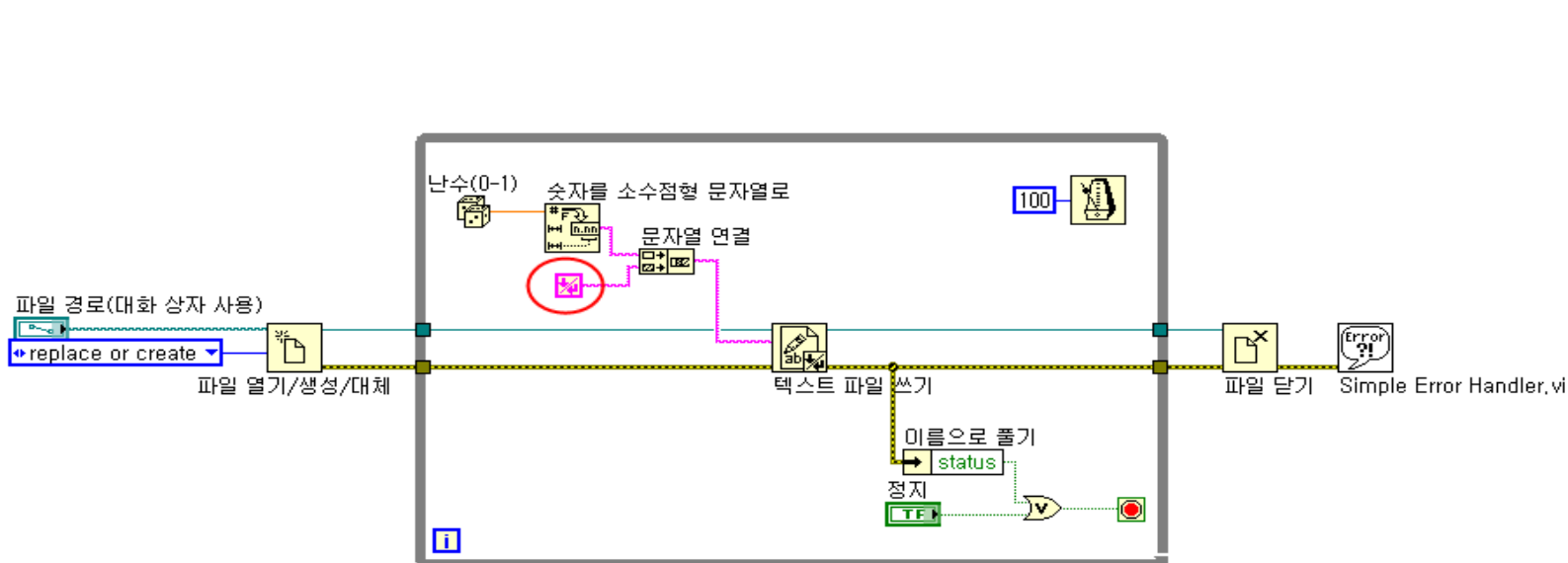


Simple Error Handler.vi



# 아스키 파일 쓰기

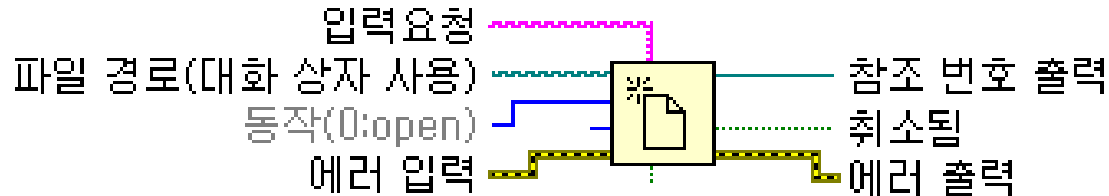
- ❖ 아스키 포맷으로 데이터를 저장하는 프로그램을 할 때 파일 열기/생성/대체.VI 에서 동작 파라미터에 'replace or create'로 설정하여 기존 파일이 있으면 대체하거나 새롭게 생성할 수 있도록 설정한다.
- ❖ 그리고 텍스트 파일에 쓰기.VI는 항상 문자열 데이터 타입만 연결할 수 있다.



ascii.txt - 메모장

파일(F)	편집(E)	서식(O)
0.247291		
0.129358		
0.969071		
0.621066		
0.029632		
0.380146		
0.741919		
0.769639		
0.169787		
0.784368		
0.487678		
0.088114		
0.436518		
0.414354		
0.458088		
0.650340		
0.726020		
0.090554		
0.185885		
0.030196		
0.385143		
0.467851		

## 파일 열기/생성/대체 [Open/Create/Replace File]



프로그램적으로 또는 파일 대화 상자를 사용하여 대화식으로 기존 파일을 열거나, 새 파일을 생성하거나, 기존 파일을 대체합니다. 이 함수는 LLB 내의 파일에는 작동하지 않습니다.

### <파일 경로(대화 상자 사용)>

- 파일에 대한 절대 경로
- 파일 경로(대화 상자 사용)를 연결하지 않으면 함수는 파일을 선택할 수 있는 대화 상자를 디스플레이
- 빈 경로 또는 상대 경로를 지정할 경우, 이 함수는 에러를 반환

### <동작>

- 수행할 작업

0 **open**(기본)—기존 파일을 엽니다.

1 **replace**—파일을 열고 파일 끝을 0으로 설정하여 기존 파일을 대체

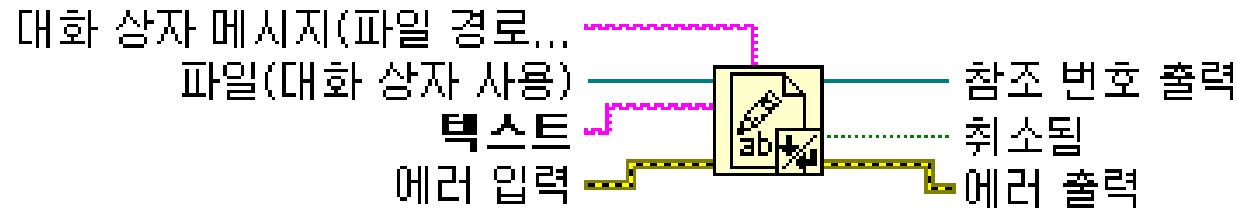
2 **create**—새 파일을 생성

3 **open or create**—기존의 파일을 열거나 존재하지 않을 경우 새 파일을 생성

4 **replace or create**—새 파일을 생성하거나 파일이 존재할 경우 파일을 대체

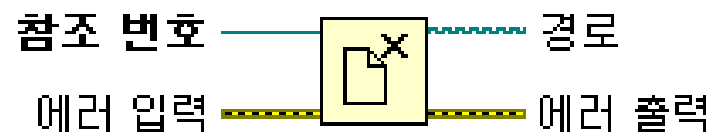
5 **replace or create with confirmation**—새 파일을 생성하거나 파일이 존재하고 권한이 주어진 경우 기존 파일을 대체.

## 텍스트 파일에 쓰기 [Write to Text File]



문자의 문자열 또는 문자열의 배열을 라인으로 파일에 씁니다. 이 함수는 LLB 내의 파일에는 작동하지 않습니다.

## 파일 닫기 [Close File]



참조 번호로 지정된 열려있는 파일을 닫고 참조 번호와 관련되어 있는 파일 경로를 반환합니다.



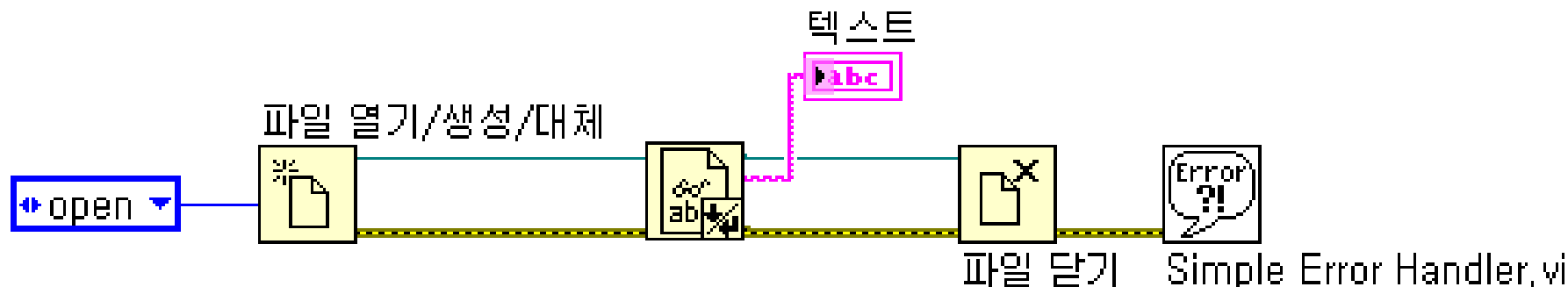


## 아스키 파일 읽기

- ❖ 주의할 점은 파일 열기/생성/대체.vi 의 동작 파라미터를 'open'으로 설정하였다.
- ❖ 읽기 프로그램은 만들어진 파일을 열어서 읽어야 하므로 쓰기 프로그램에서 'replace or create'로 설정했을 때 오는 차이를 보입니다.
- ❖ 그리고 텍스트 파일로부터 읽기.VI 에서 읽어온 데이터 타입은 항상 문자열이다.

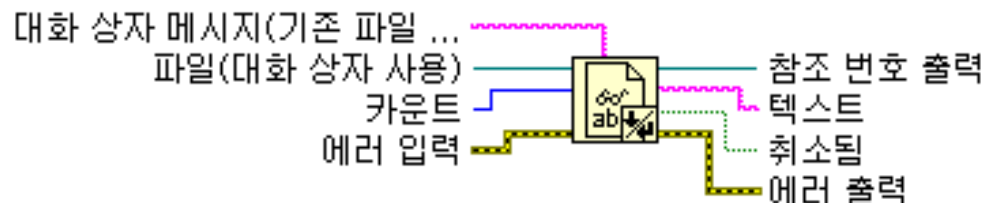


# 아스키 파일 읽기



## 기본 도움말

### 텍스트 파일로부터 읽기 [Read from Text File]



지정된 문자의 개수 또는 바이트 스트림 파일로부터 라인을 읽습니다. 이 함수는 LLB 내의 파일에는 작동하지 않습니다.

### 텍스트

0,359667  
0,362184  
0,986492  
0,373530  
0,639041  
0,550783  
0,911672  
0,267101  
0,645492  
0,401305  
0,857509  
0,933686  
0,155898  
0,325329  
0,130470  
0,157404  
0,278737  
0,223075

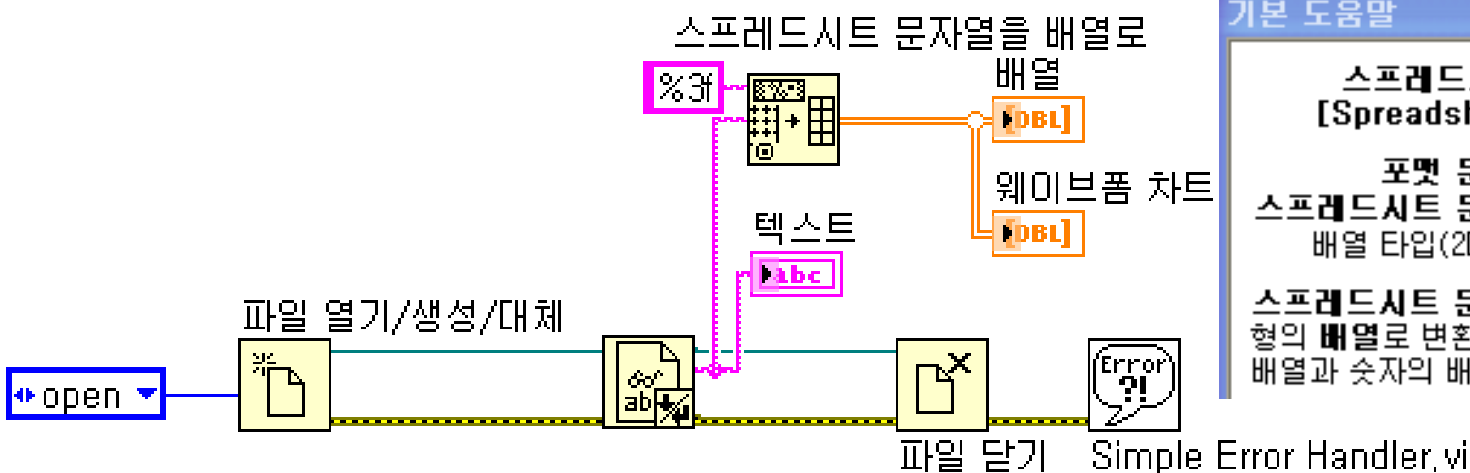


## 아스키 파일 읽기

- ❖ 아스키 포맷으로 저장된 파일로부터 읽어온 문자열 데이터를 웨이브폼 차트를 통해 보려면 그림과 같이 문자열 데이터를 숫자형 배열로 변경해야 한다.



# 아스키 파일 읽기

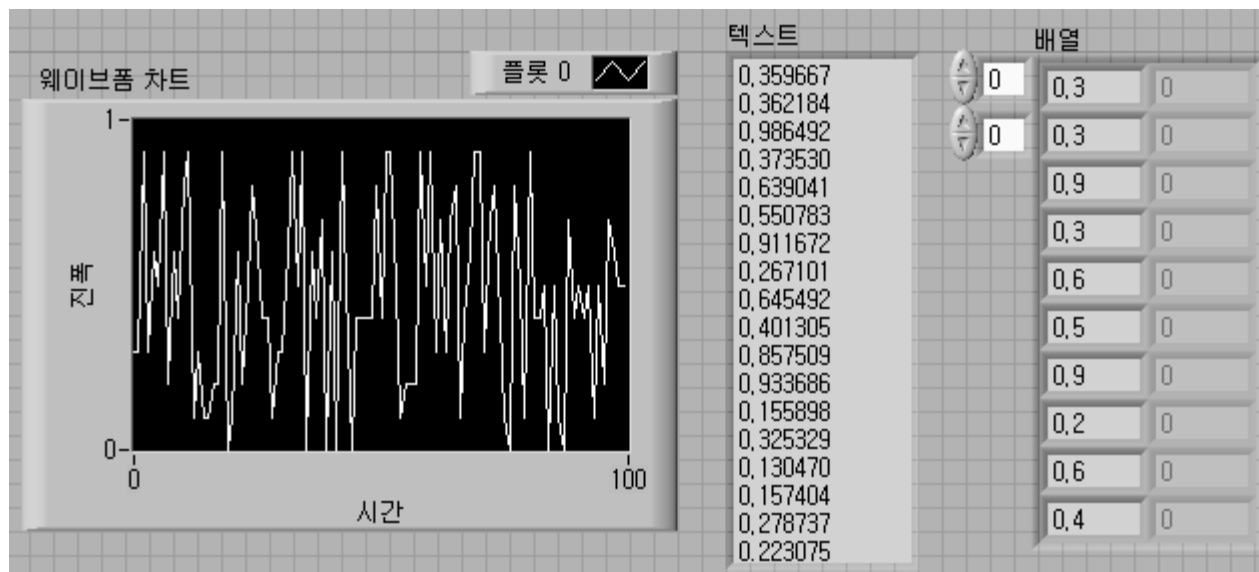


## 기본 도움말

### 스프레드시트 문자열을 배열로 [Spreadsheet String To Array]

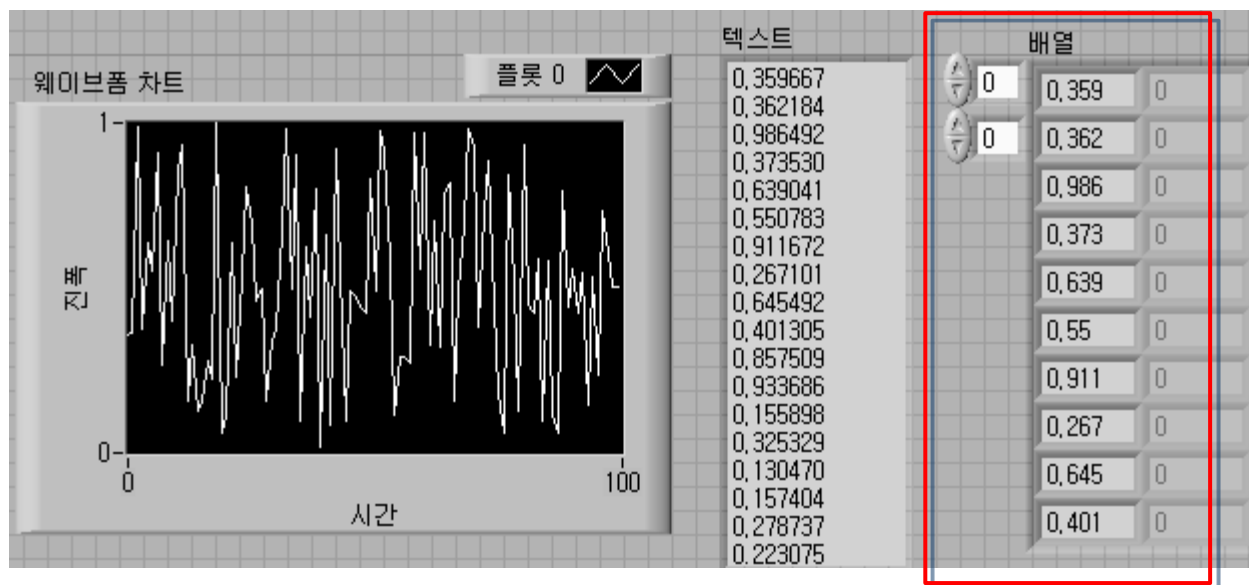
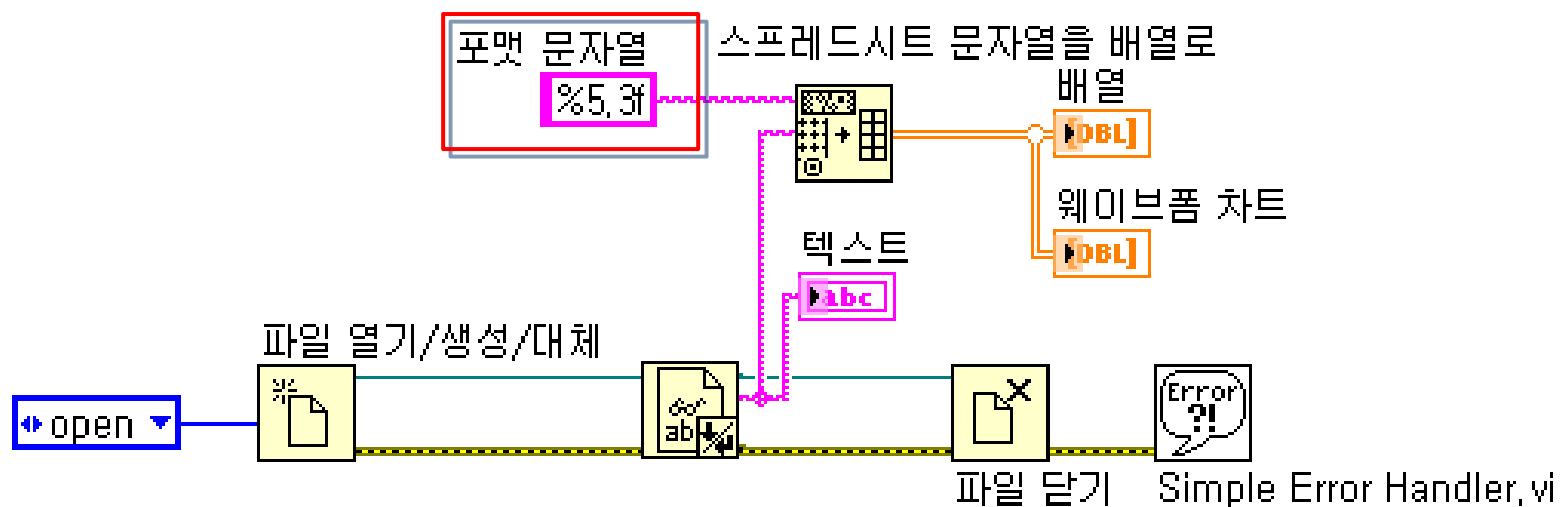
포맷 문자열  
스프레드시트 문자열  
배열 타입(2D Dbl)

스프레드시트 문자열을 배열 타입의 차원과 형의 배열로 변환합니다. 이 함수는 문자열의 배열과 숫자의 배열에서 작동합니다.



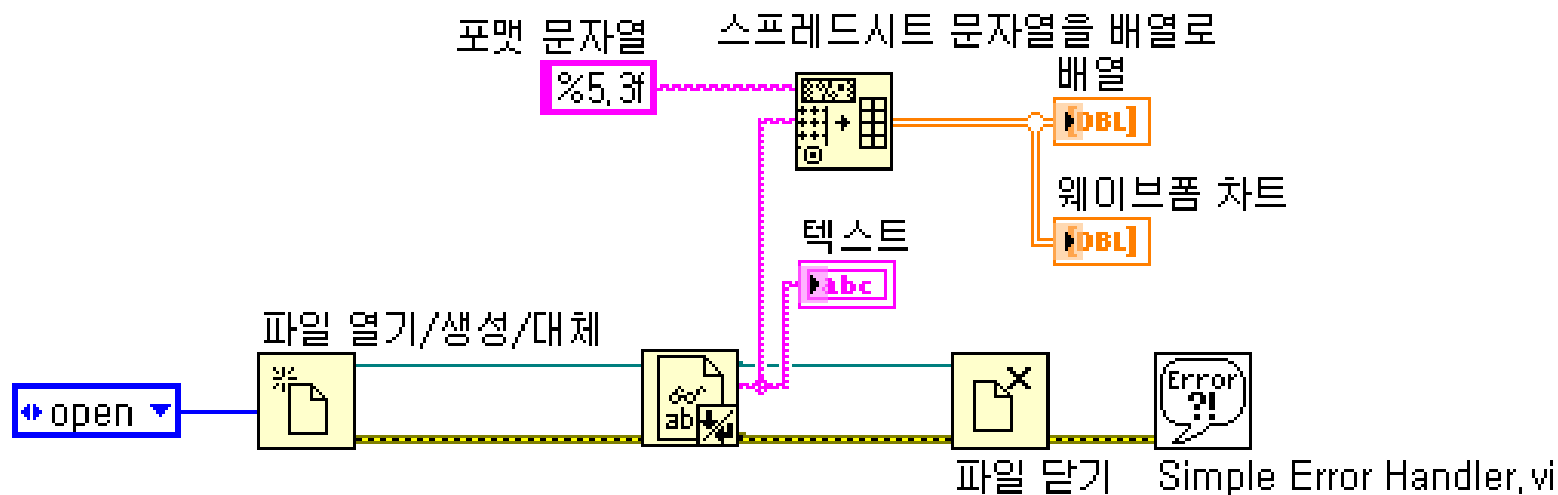


# 아스키 파일 읽기





# 아스키 파일 읽기



## 기본 도움말

### 스프레드시트 문자열을 배열로 [Spreadsheet String To Array]

포맷 문자열  
스프레드시트 문자열  
배열 타입(2D dbl)

스프레드시트 문자열을 배열 타입의 차원과 형의 배열로 변환합니다. 이 함수는 문자열의 배열과 숫자의 배열에서 작동합니다.

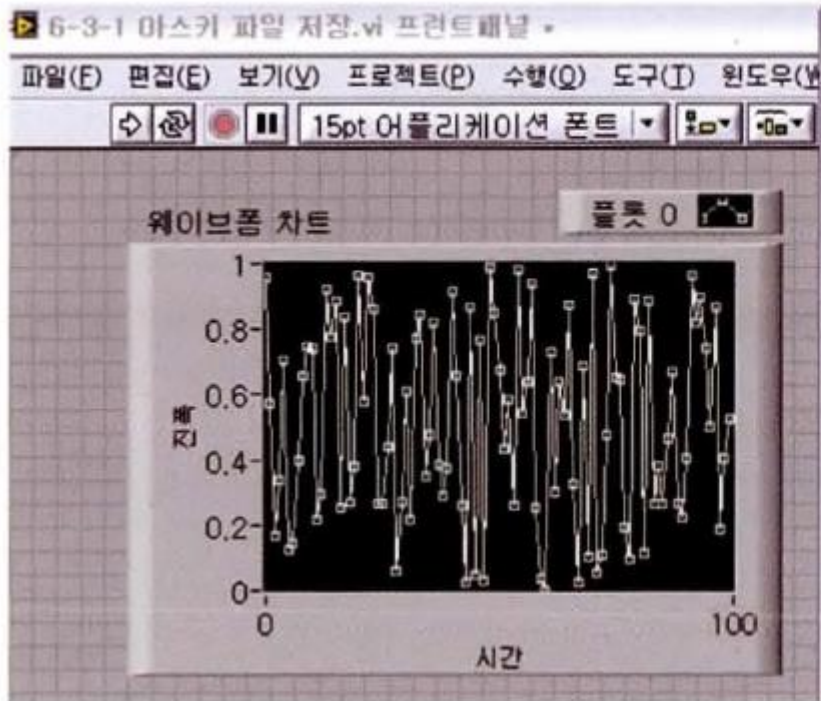
-텍스트 파일을 숫자로 읽을 때 tab키는 가로 행의 원소를 분리하고  
, enter키는 행의 줄바꿈을 한다

## 실습6-3-1) 아스키 파일 쓰기



## 실습6-3-1) 아스키 파일 쓰기

- ❖ 새 VI를 열고, 그림과 같이 블록다이어그램을 구성
- ❖ 실행을 하면 0에서 1 사이의 임의의 숫자 100개가 웨이브폼 차트에 출력된다.

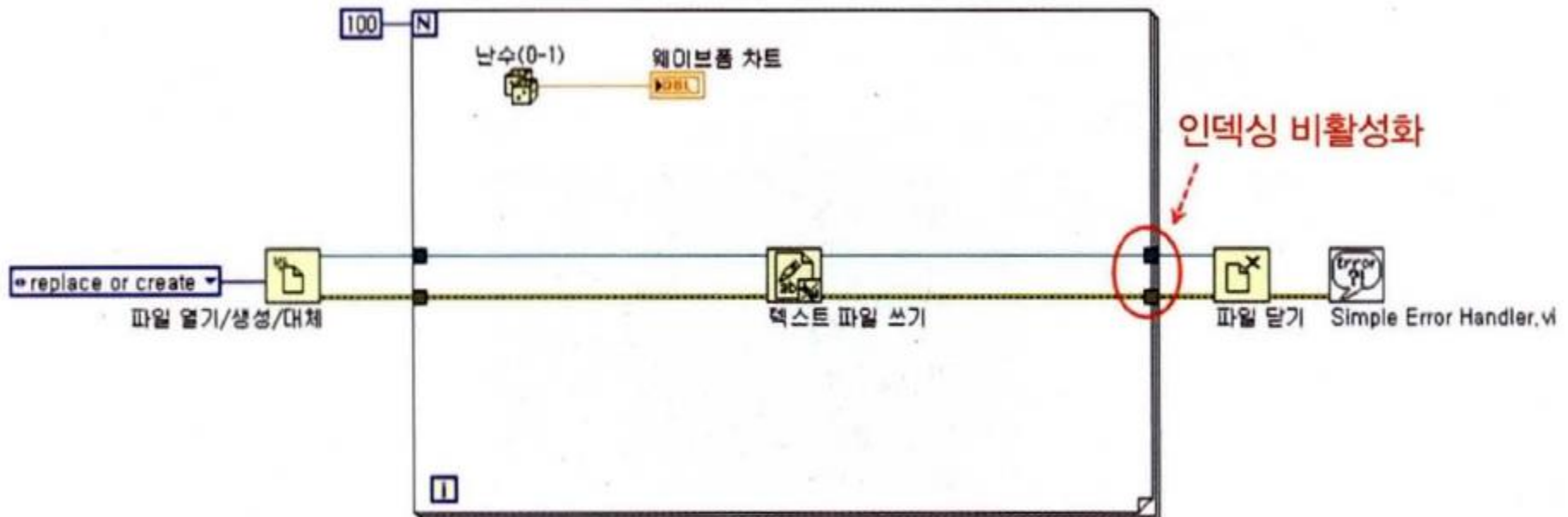






## 실습6-3-1) 아스키 파일 쓰기

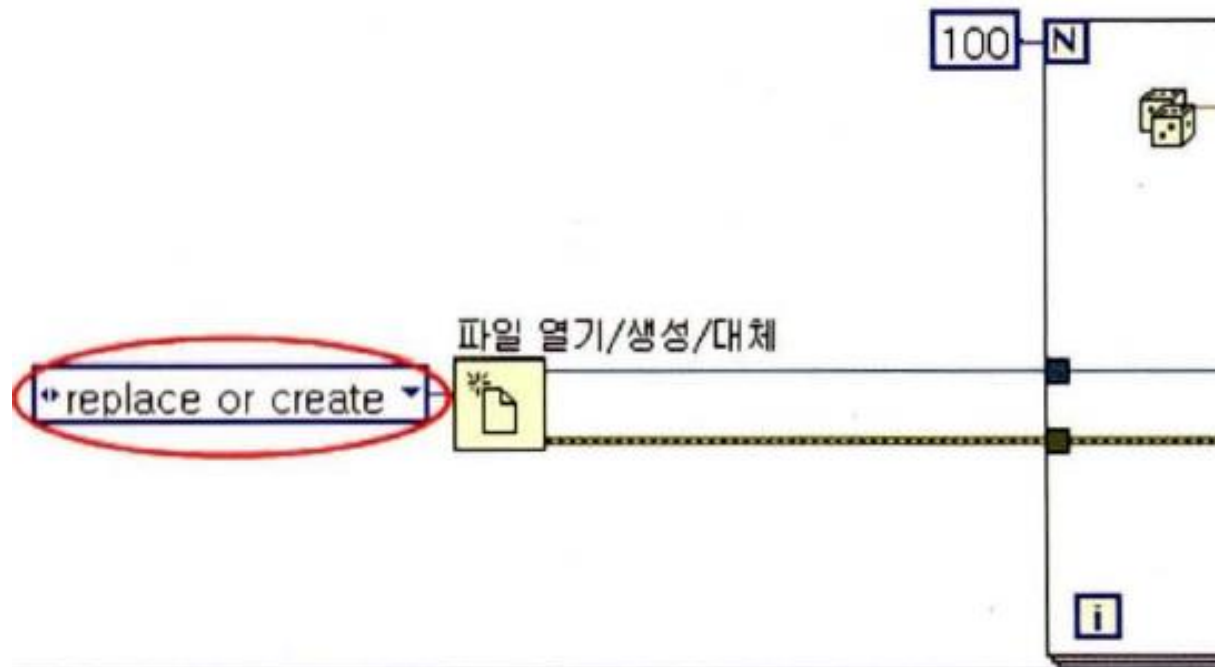
- ❖ 블록다이어그램에 파일 열기/생성/대체.vi, 텍스트 파일에 쓰기.VI, 파일 닫기.VI, 단순 에러 핸들러.VI를 추가하여 그림과 같이 수정한다





## 실습6-3-1) 아스키 파일 쓰기

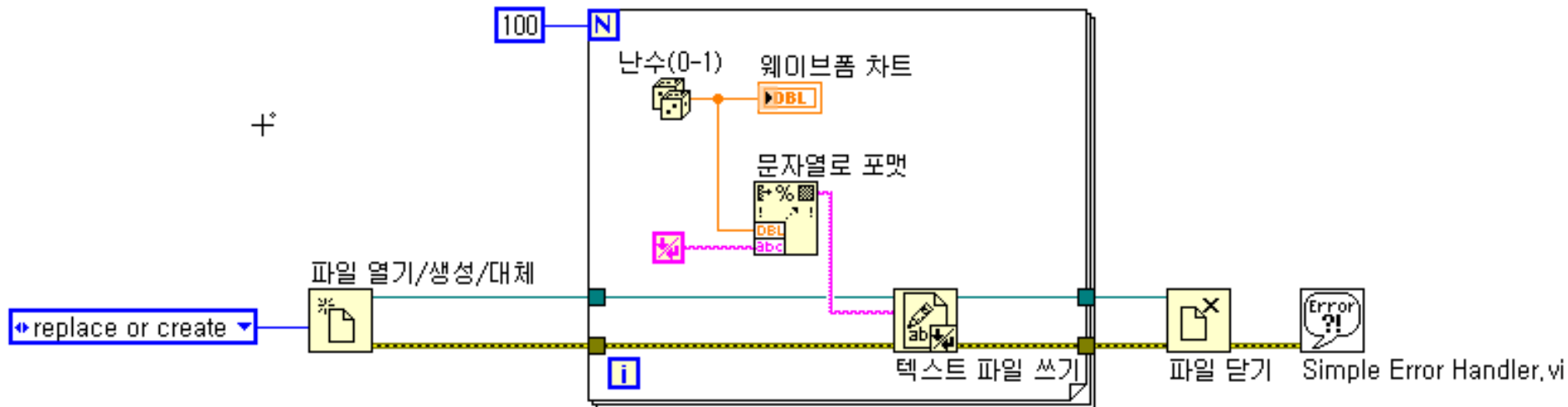
- ❖ 파일 열기/생성/대체 .VI의 동작 파라미터에서 바로가기 메뉴 > 생성 > 상수를 선택하여 'replace or create'로 설정

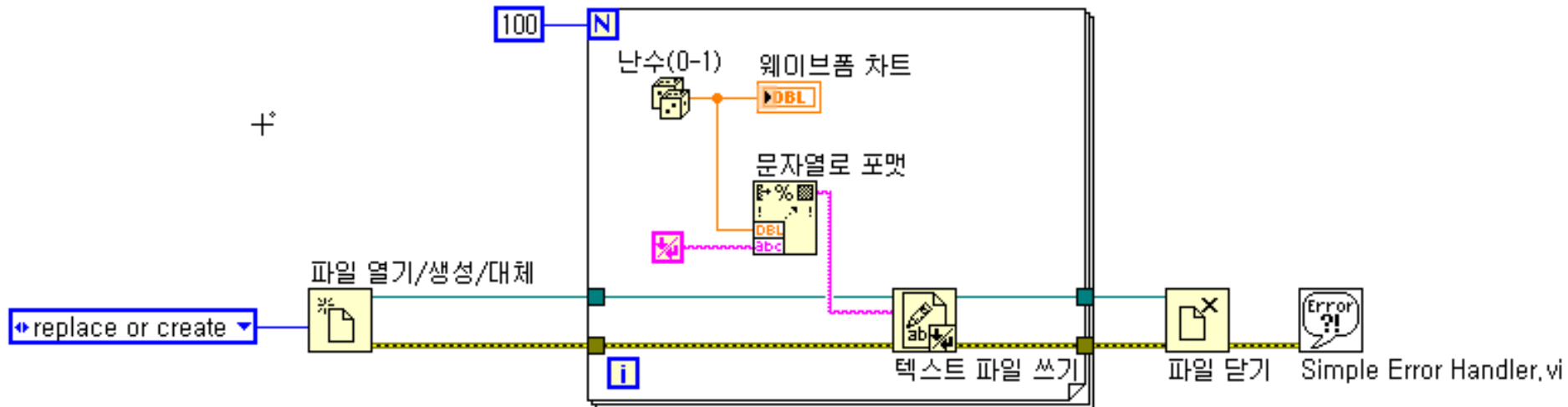




## 실습6-3-1) 아스키 파일 쓰기

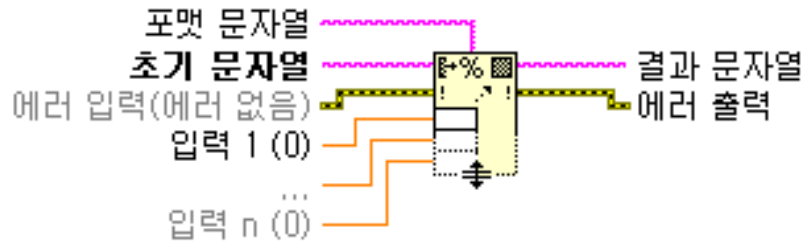
- ❖ 문자열로 포맷.vi를 사용하여 숫자형을 문자열로 변경하고 라인 끝 상수와 연결된 출력을 텍스트 파일에 쓰기.VI의 텍스트 파라미터에 그림과 같이 입력한다.





## 기본 도움말

### 문자열로 포맷 [Format Into String]



포맷 문자열 경로, 열거형 타입, 타임스탬프, 불리언,  
또는 텍스트의 숫자 데이터.

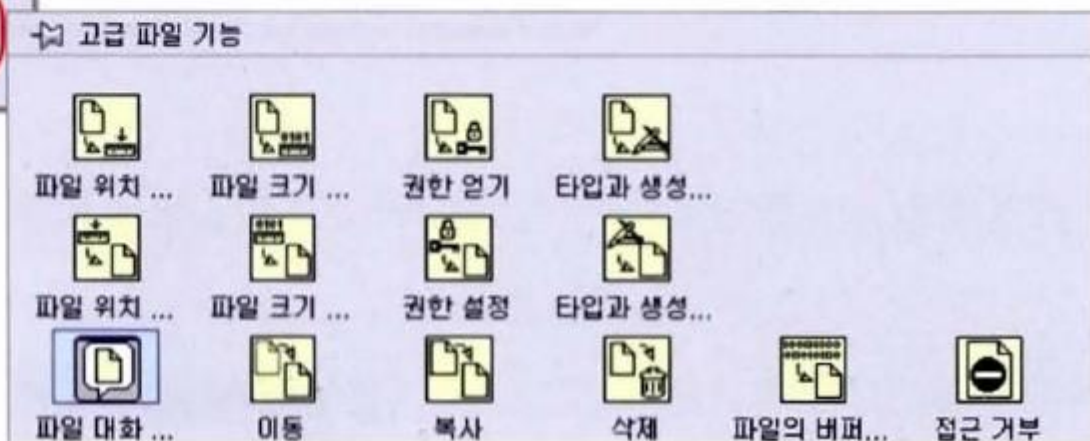
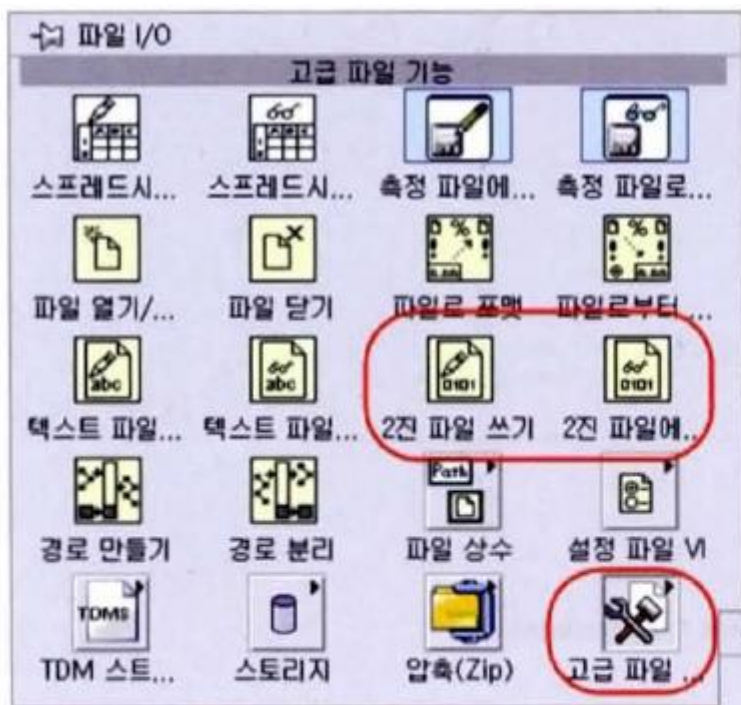
### <문자열로 포맷>

-숫자를 문자로 만들고 입력된 값들을 연결  
해주는 두 가지 역할을 한꺼번에 해주는 함수



# 바이너리 파일 쓰기

❖ 바이너리는 다양한 노드들을 이용하여 프로그램이 가능





## 바이너리 파일 쓰기

- ❖ 바이너리 포맷은 아스키 포맷과는 달리 어떠한 데이터 타입도 데이터 변환 없이 2진 파일에 쓰기.VI에 바로 연결
- ❖ 2진 파일에 쓰기.VI에서 배열 또는 문자열 크기를 앞에 추가? 파라미터는 2진 파일에 쓰기.vi에 연결되는 데이터 타입이 배열 또는 문자열인 경우 '참' 값을 설정하면 이들의 크기를 자동으로 추가해준다.
- ❖ 바이너리 포맷으로 저장된 파일을 그림과 같이 메모장에서 열어서 보면 데이터가 깨져보인다.
- ❖ 이는 문자열로 데이터 변환없이 원래의 데이터 타입 그대로 저장되기 때문이다



# 바이너리 파일 쓰기

파일 경로(대화 상자 사용)  
D:\EduVIEW\book\VIEW\binary.dat

데이터  
0  
0.277602  
0.580057  
0.794935  
0.216942  
0.963619

binary.dat - 메모장  
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)  
???쵸??퀵뵤???궤???污쟁C????옴X?遠xxQ??甸쟁\_=?a랏

100 N  
난수(0-1)  
데이터 DBL  
데이터 변환 없이 연결  
replace or create  
파일 열기/생성/대체  
2진 파일 쓰기  
파일 닫기  
Simple Error Handler.vi

배열 또는 문자열 크기를 앞에 추가? : 연결되는 데이터 타입이 배열 또는 문자열인 경우, '참'값을 설정하면 이들의 크기를 자동으로 추가



# 바이너리 파일 쓰기

## 기본 도움말

### 2진 파일에 쓰기 [Write to Binary File]



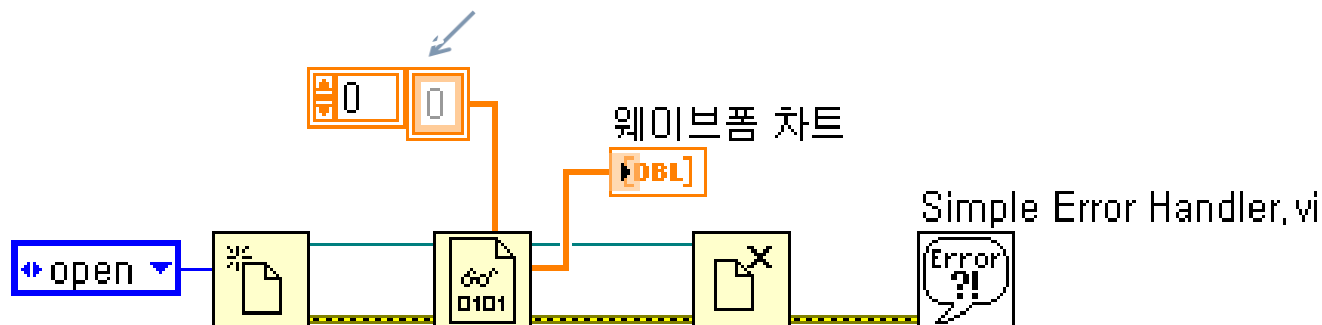
새 파일에 2진 데이터를 쓰거나, 기존 파일에 데이터를 추가하거나, 파일의 내용을 대체합니다. 이 함수는 LLB 내의 파일에는 작동하지 않습니다.





# 바이너리 파일 읽기

읽어 올 데이터 타입을 선언 해 주어야 함



## 기본 도움말

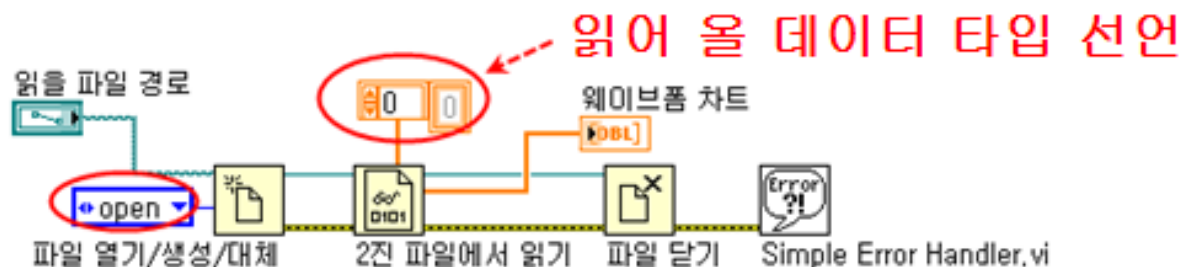
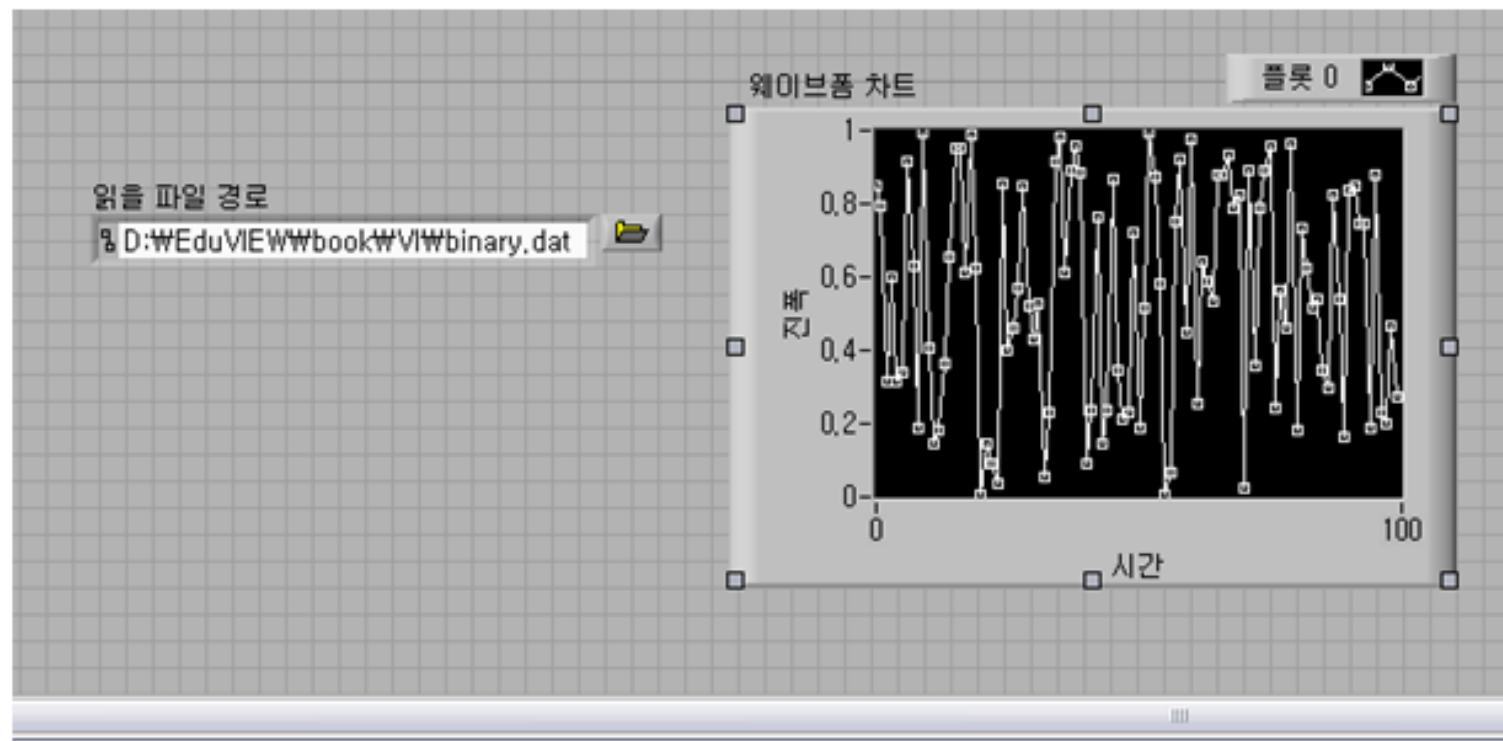
### 2진 파일로부터 읽기 [Read from Binary File]



파일로부터 2진 데이터를 읽고 **데이터**에서 그것을 반환합니다.  
데이터를 읽는 방법은 지정된 파일의 포맷에 달려있습니다. 이 함수는 LLB 내의 파일에는 작동하지 않습니다.



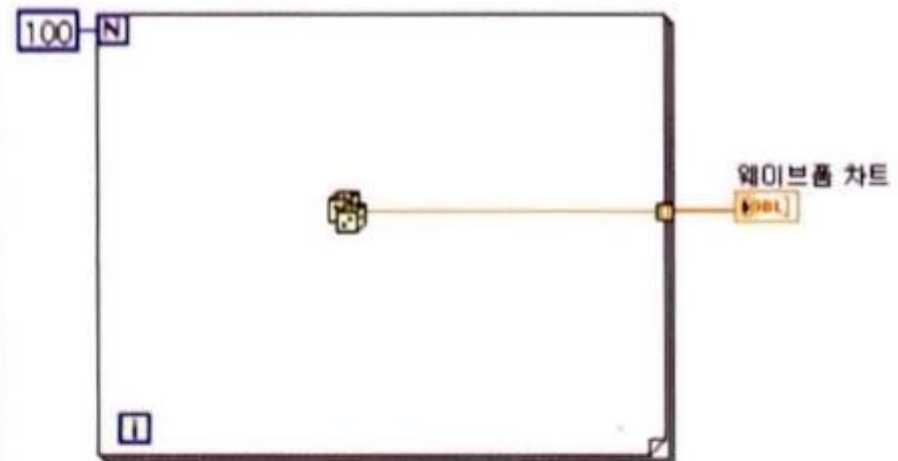
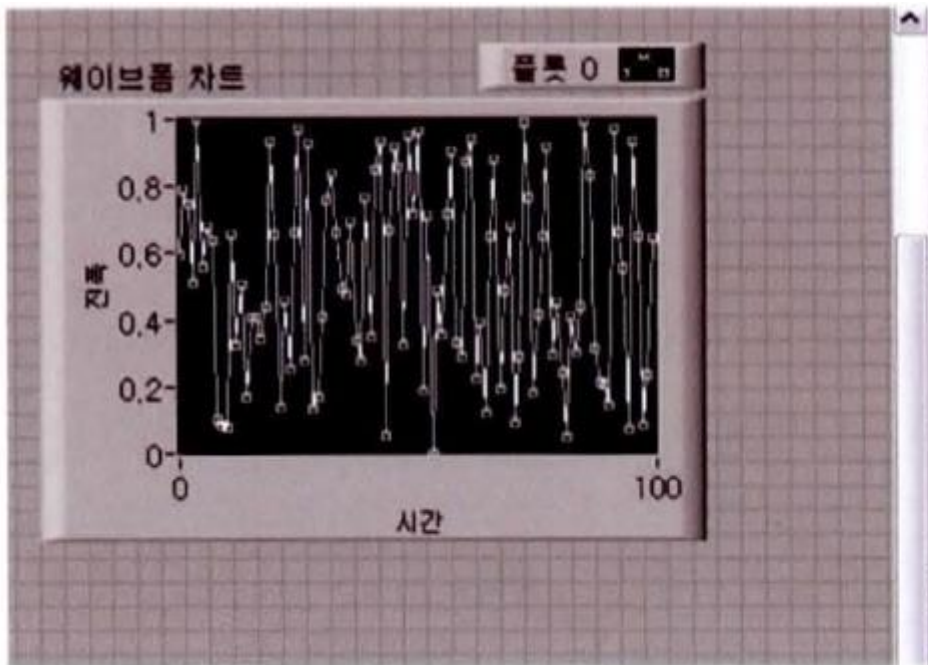
# 바이너리 파일 읽기



## 실습6-4-1) 바이너리 파일 쓰기



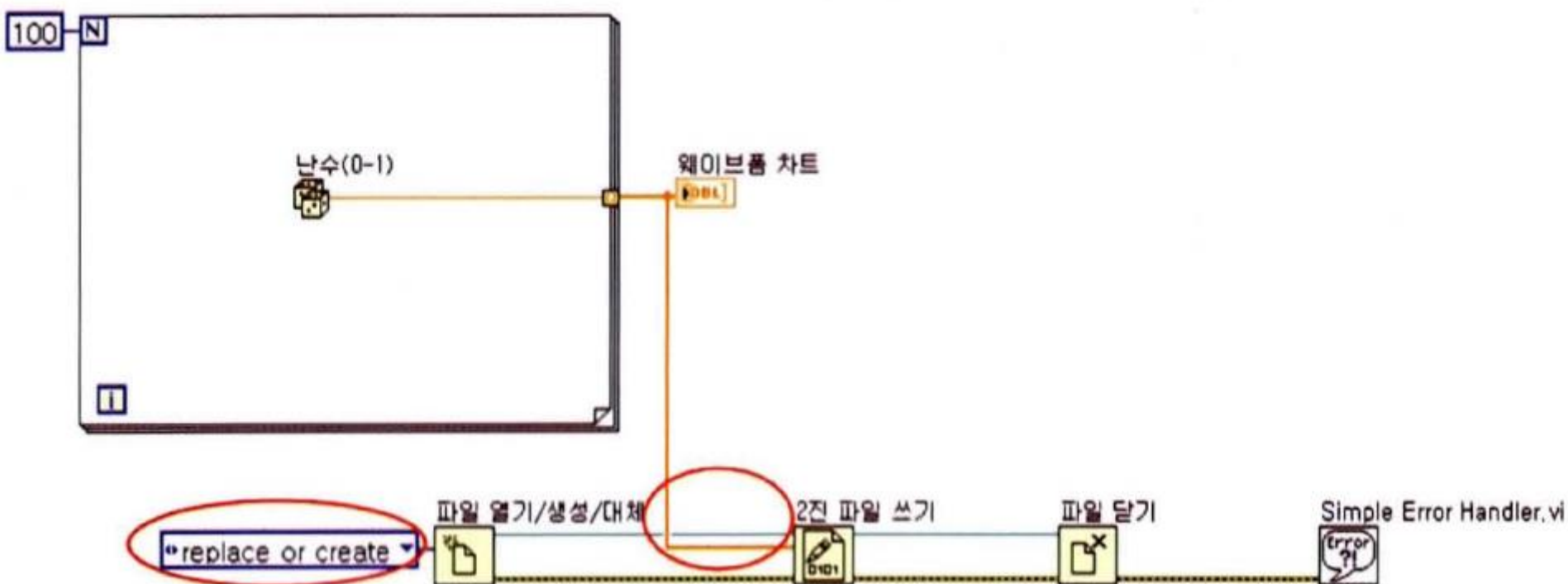
## 실습6-4-1) 바이너리 파일 쓰기





## 실습6-4-1) 바이너리 파일 쓰기

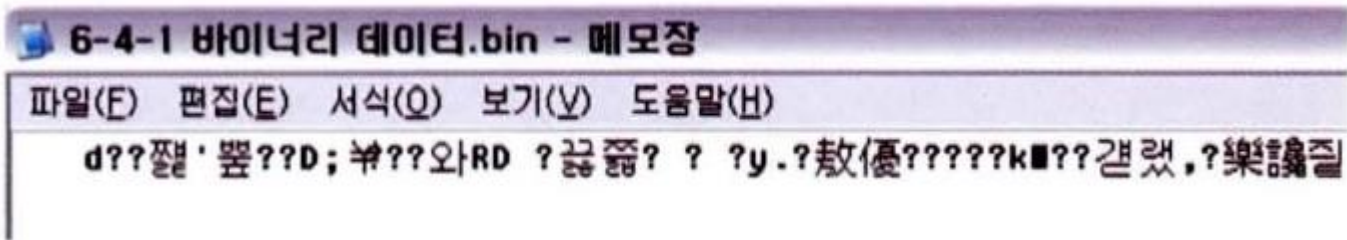
- ❖ 블록다이어그램에 파일 열기/생성/대체.vi, 2진 파일에 쓰기.VI, 파일 닫기.vi, 단순 에러 핸들러.VI를 추가하여 그림과 같이 수정.
- ❖ 파일 열기/생성/대체.vi의 동작 파라미터에서 바로가기 메뉴 > 생성 > 상수를 선택한다.





## 실습6-4-1) 바이너리 파일 쓰기

- ❖ 실행을 하면 파일 경로를 입력하는 창이 나타나고 파일명을 입력해 주면 바이너리 파일이 생성된다.
- ❖ 이 파일을 메모장에서 확인해 보면 그림과 같이 깨져 있다.

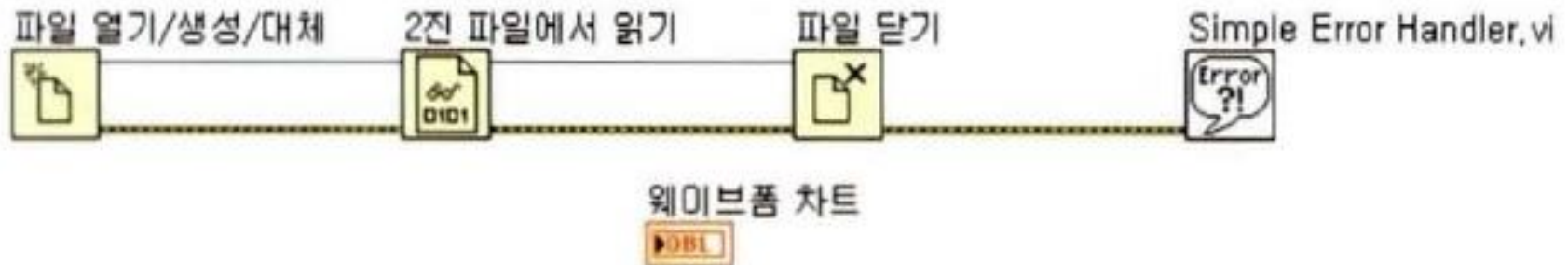


## 실습6-4-2) 바이너리 파일 읽기



## 실습6-4-2) 바이너리 파일 읽기

- ❖ 새 VI를 열고 프런트패널에는 웨이브폼 차트를, 블록다이어그램에는 파일 열기/생성/대체.vi, 2진 파일로부터 읽기.vi, 파일 닫기.VI, 단순 에러 핸들러.VI를 추가하여 그림과 같이 구성한다.







## 실습6-4-2) 바이너리 파일 읽기

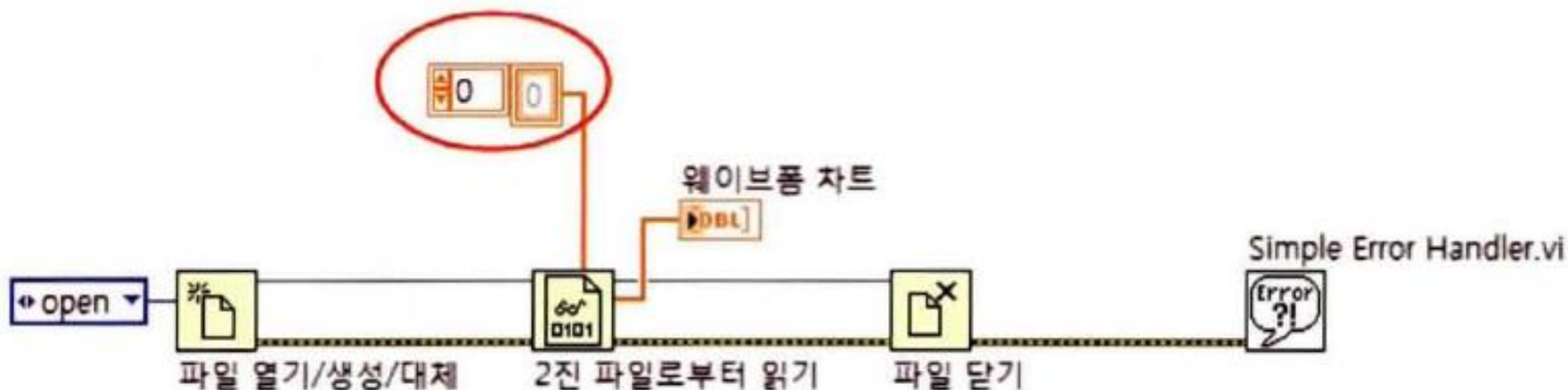
- ❖ 블록다이어그램에서 파일 열기/생성/대체 .vi의 동작 파라미터에서 바로가기메뉴 > 생성 > 상수를 선택.
- ❖ 그리고 그림과 같이 'open'으로 설정





## 실습6-4-2) 바이너리 파일 읽기

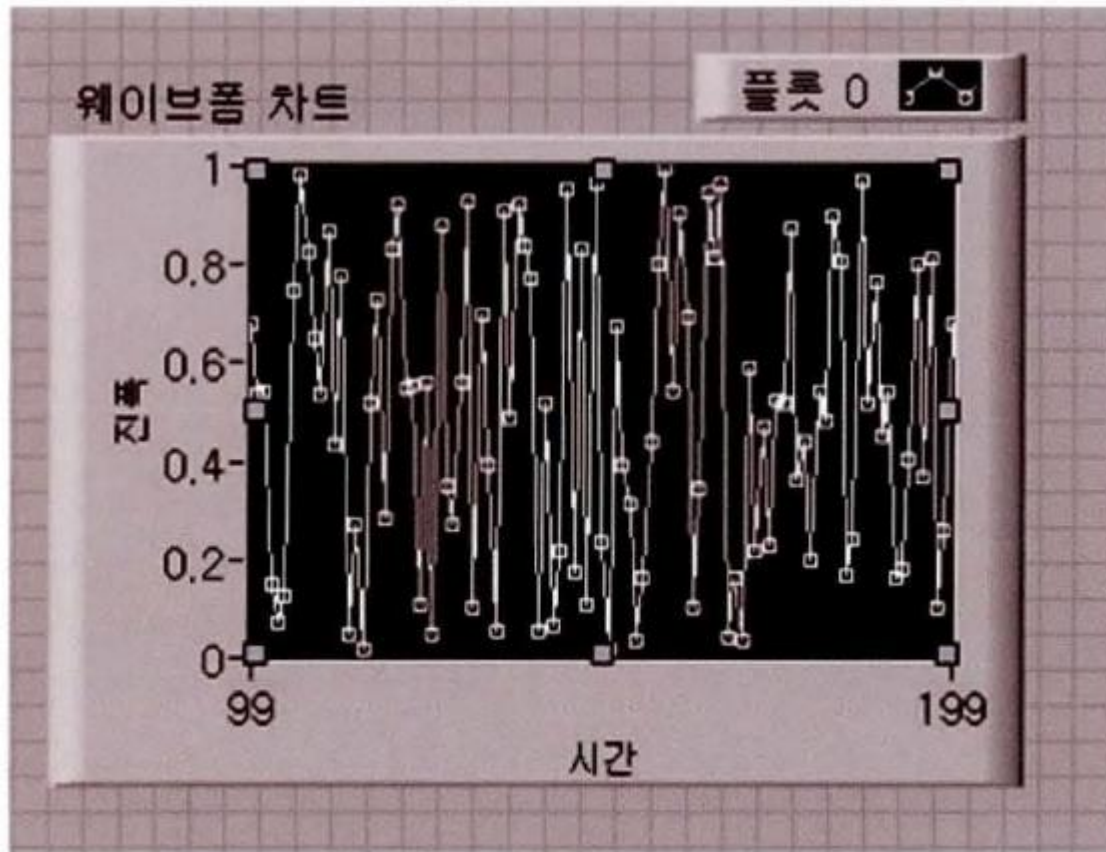
- ❖ 2진 파일로부터 읽기.vi의 데이터 타입 파라미터에 '실습 6.4.1'에서 저장한 데이터 타입인 배정도 배열을 상수로 만들어 연결.
- ❖ 다음으로 2진 파일로부터 읽기.VI의 데이터 파라미터에 웨이브폼 차트를 연결한다.





## 실습6-4-2) 바이너리 파일 읽기

- ❖ 실행하고 '실습 6.4. 1'에서 생성한 파일을 지정해 주면 데이터가 웨이브폼 차트에 출력





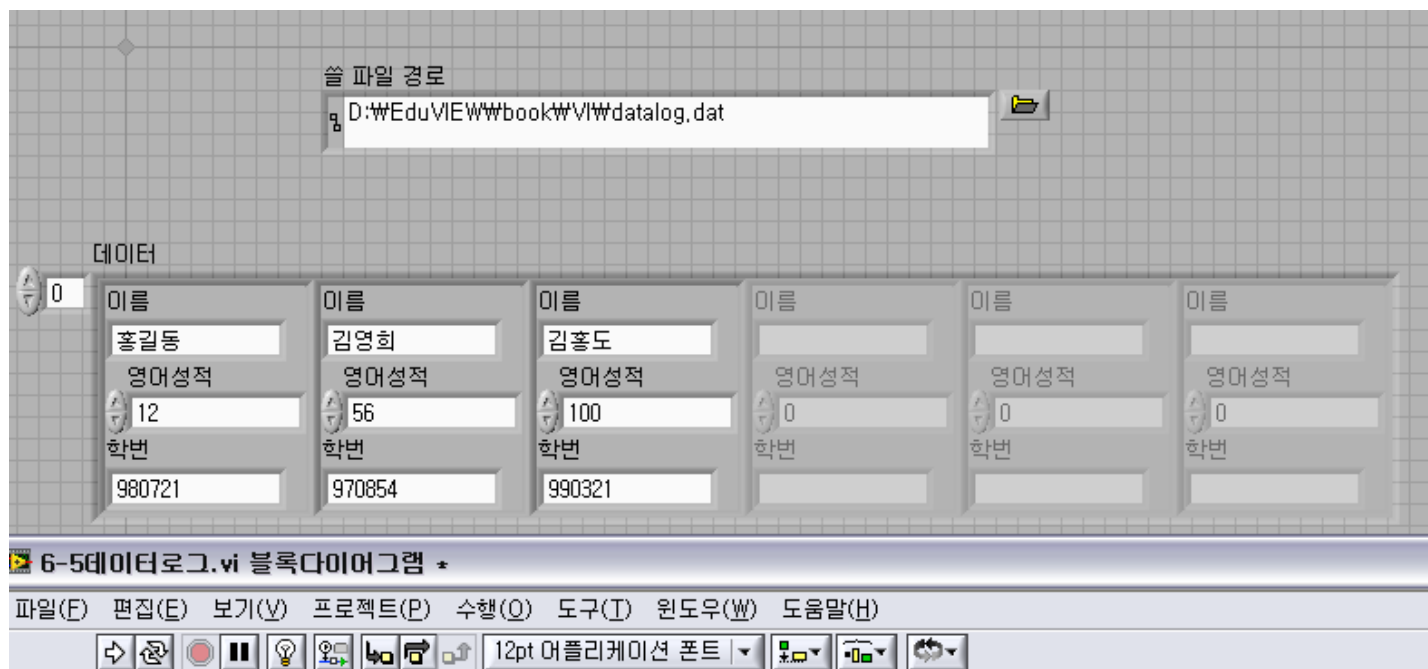
## 데이터로그 파일 쓰기

- ❖ 데이터로그 파일은 클러스터 데이터를 파일에 쓰기 또는 읽기를 쉽게 해준다
- ❖ 그림은 데이터 로그전용 노드들이다.



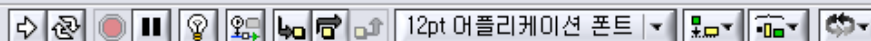


# 데이터로그 파일 쓰기



6-5데이터로그.vi 블록다이어그램 \*

파일(E) 편집(E) 보기(V) 프로젝트(P) 수행(Q) 도구(T) 윈도우(W) 도움말(H)





# 데이터로그 파일 읽기

읽을 파일 경로  
D:\EduVIEW\book\VI\W\datalog.dat

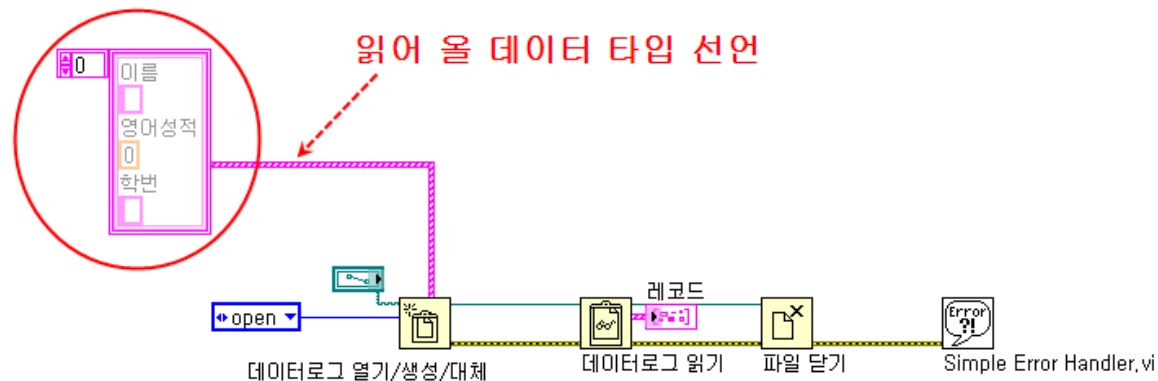
레코드

이름	이름	이름	이름	이름
홍길동	김영희	김홍도	영어성적	영어성적
영어성적	영어성적	영어성적	영어성적	영어성적
12	56	100	0	0
학번	학번	학번	학번	학번
980721	970854	990321		

로그.vi 블록다이어그램 \*

(E) 보기(V) 프로젝트(P) 수행(Q) 도구(T) 윈도우(W) 도움말(H)

12pt 어플리케이션 폰트



## **실습6-5-1) 데이터로그 파일 쓰기**



## 실습6-5-1) 데이터로그 파일 쓰기

❖ 프런트패널에 있는 클러스터 배열을 저장하는 프로그램을 만든다.

데이터

이름	이름	이름	이름	이름	이름
홍길동	유관순	이영이	박영수	김철수	
영어성적	영어성적	영어성적	영어성적	영어성적	영어성적
67	85	33	27	100	0
학번	학번	학번	학번	학번	학번
87654	56372	13462	34568	57489	

저장 경로

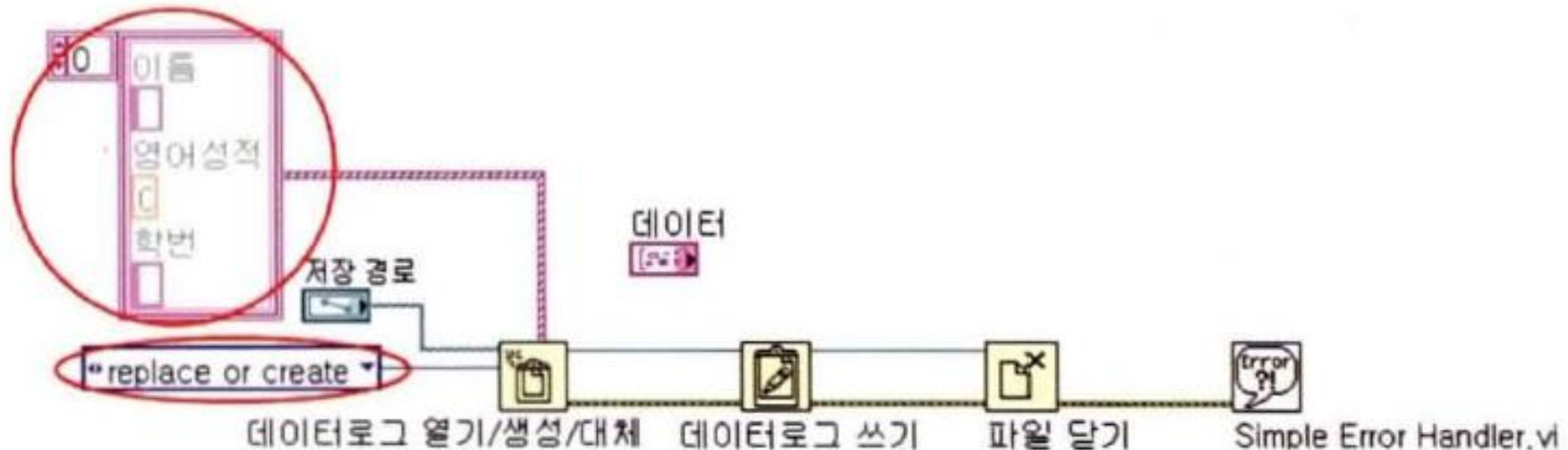
c:\Wtest.dat





## 실습6-5-1) 데이터로그 파일 쓰기

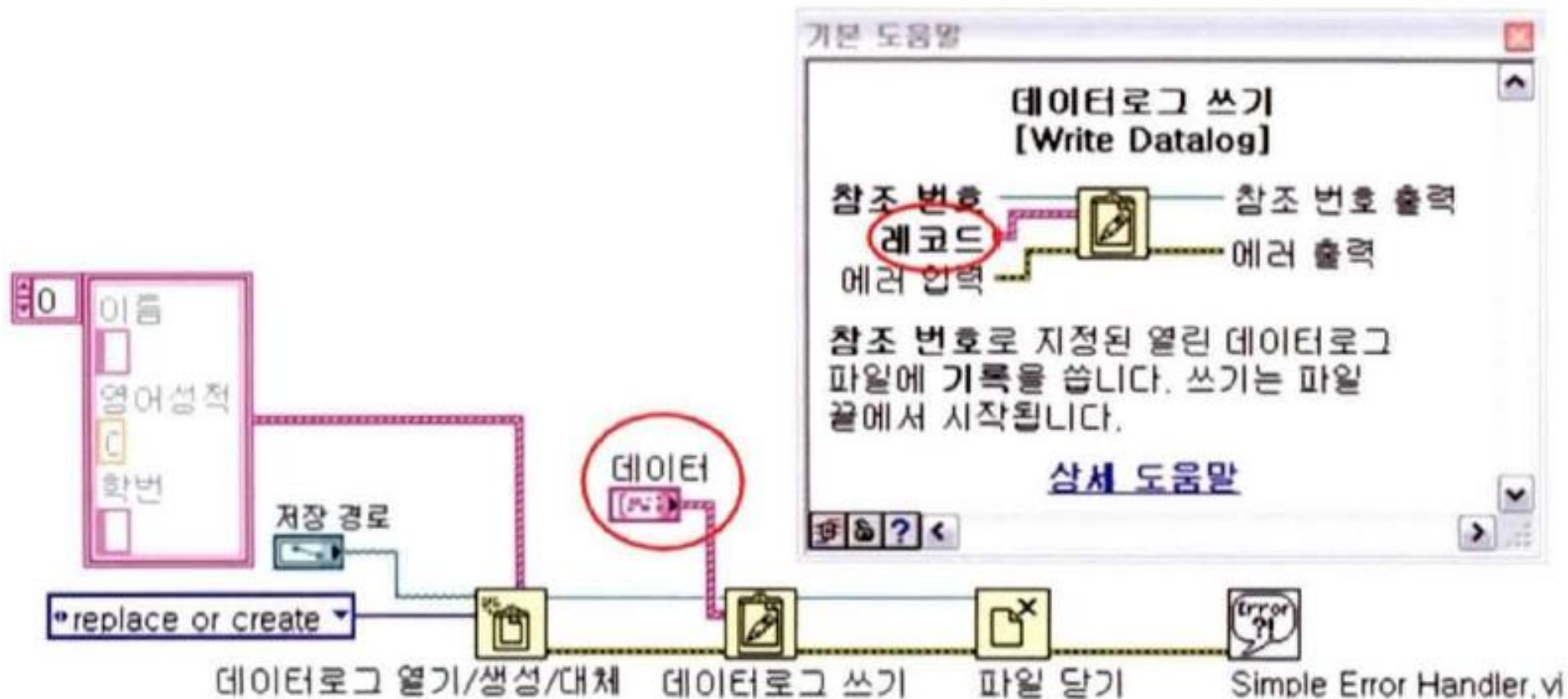
- ❖ 블록다이어그램에 데이터로그 열기/생성/대체.vi, 데이터로그 쓰기.VI, 파일 닫기.vi, 단순 에러 핸들러.VI를 추가하여 그림과 같이 구성한다.
- ❖ 데이터로그 열기/생성/대체.vi의 동작 파라미터에서 바로가기메뉴 > 생성 > 상수를 선택하여 'replace or create'로 설정하고, 레코드 타입 파라미터에는 저장할 데이터 타입을 선언한다.





## 실습6-5-1) 데이터로그 파일 쓰기

- ❖ 데이터로그 쓰기.vi의 레코드 파라미터에 '데이터' 터미널을 연결하고 실행을 하면 C:\Wtest.dat 파일이 생성



# 실습6-5-2) 데이터로그 파일 읽기



## 실습6-5-2) 데이터로그 파일 읽기

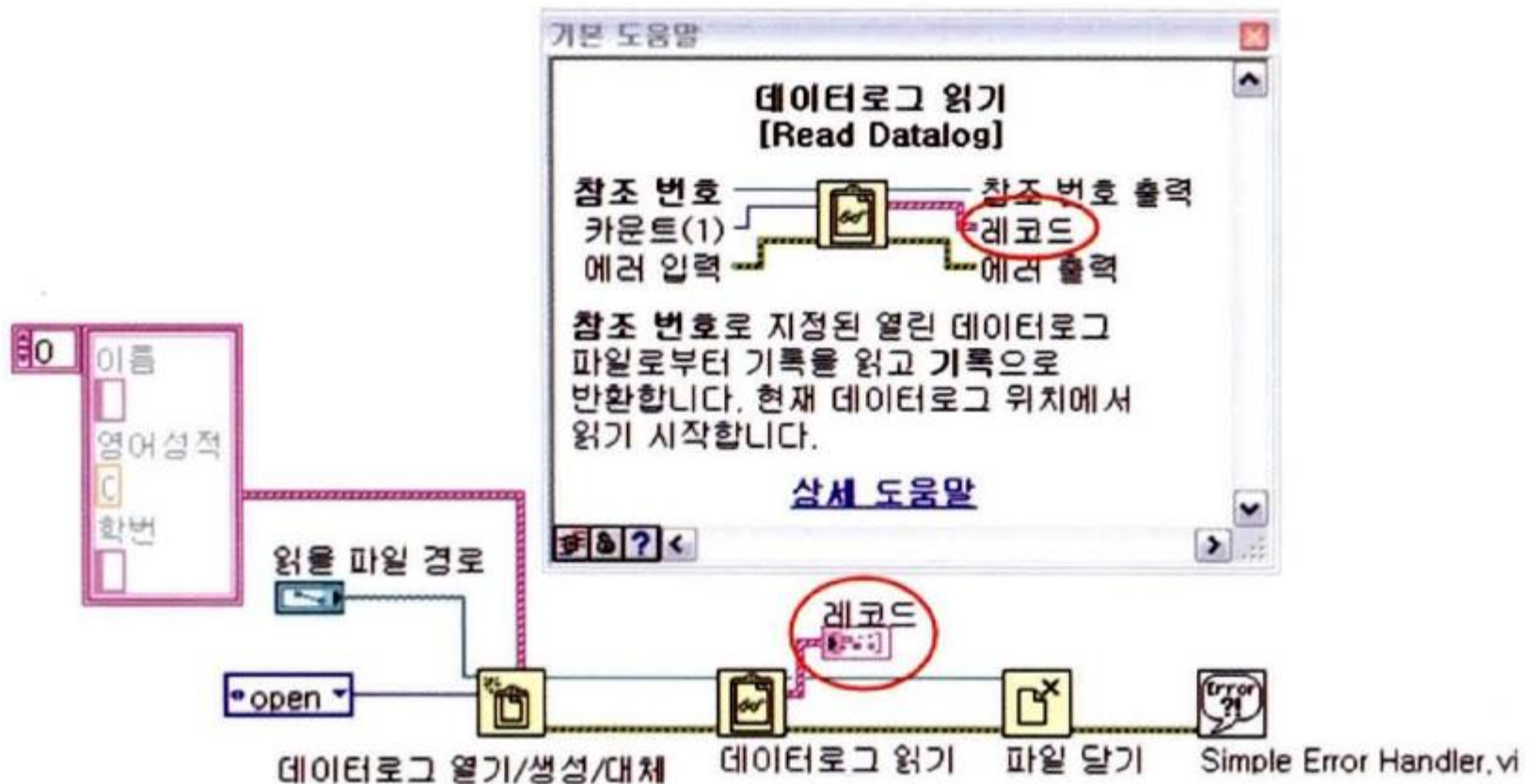
- ❖ 블록다이어그램에 데이터로그 열기/생성/대체.vi, 데이터로그 읽기.vi, 파일 닫기.vi, 단순 에러 핸들러.vi를 추가하여 그림과 같이 구성





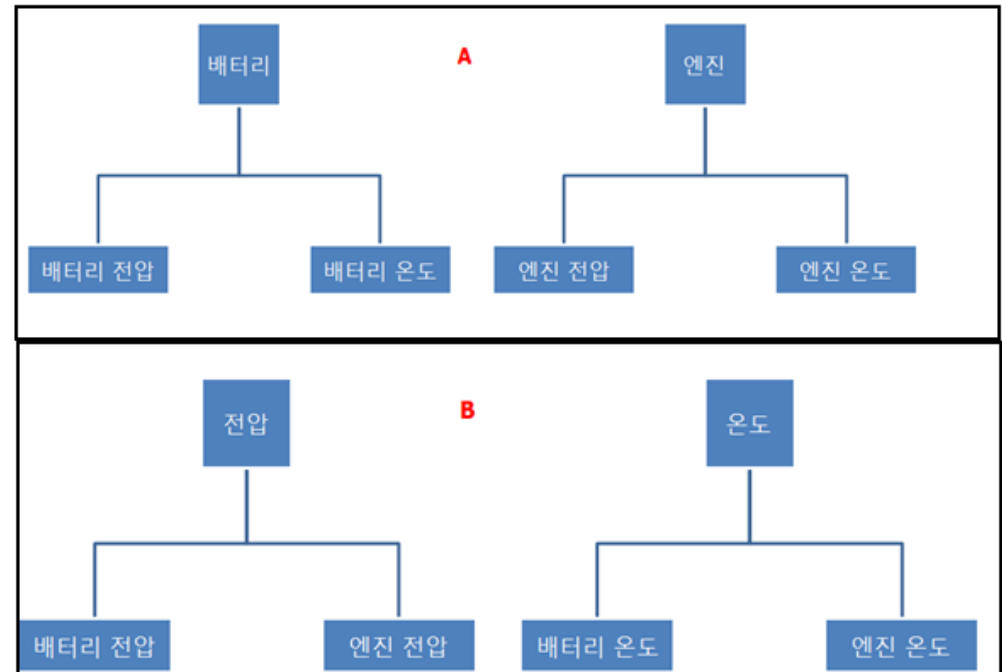
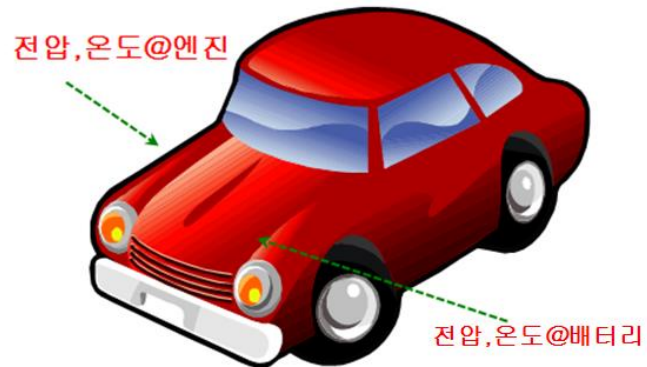
## 실습6-5-2) 데이터로그 파일 읽기

- ❖ 데이터로그 읽기.VI의 레코드 파라미터에 '레코드' 터미널을 연결하고 실행하여 C:\wtestl.dat 파일을 읽어온다.





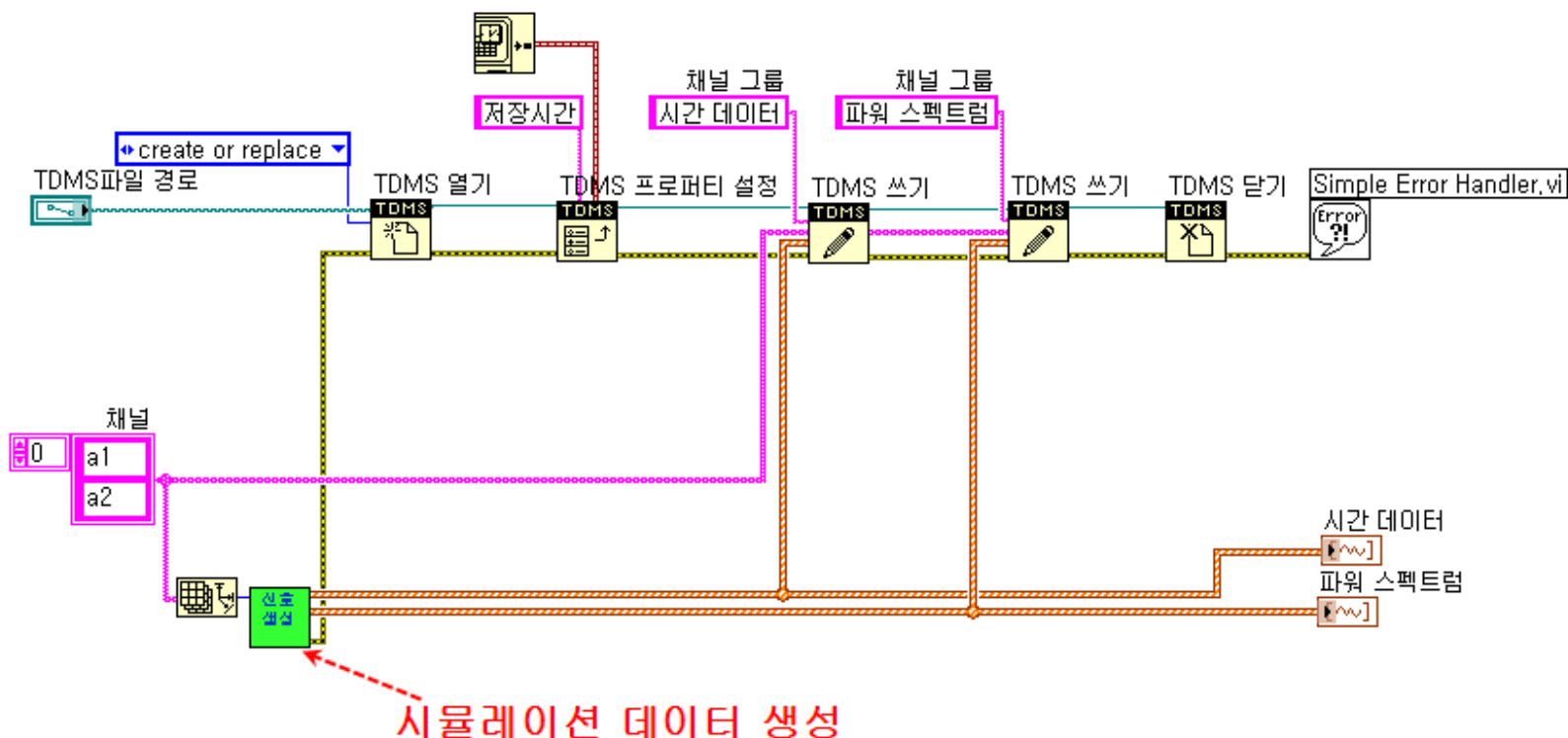
# TDMS



- 간단한 쿼리 적용하여 파일을 읽어 올 수 있어 데이터 수집과 함께 사용하면 유용함.
- 데이터 저장 할 때, '채널'과 '채널 그룹'을 지정함.



# TDMS 파일 쓰기



# 실습6-6-1) TDMS 파일 쓰기

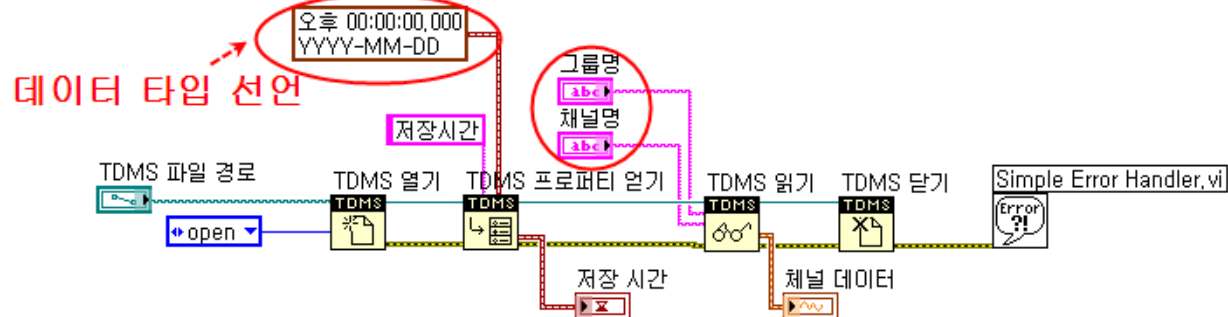
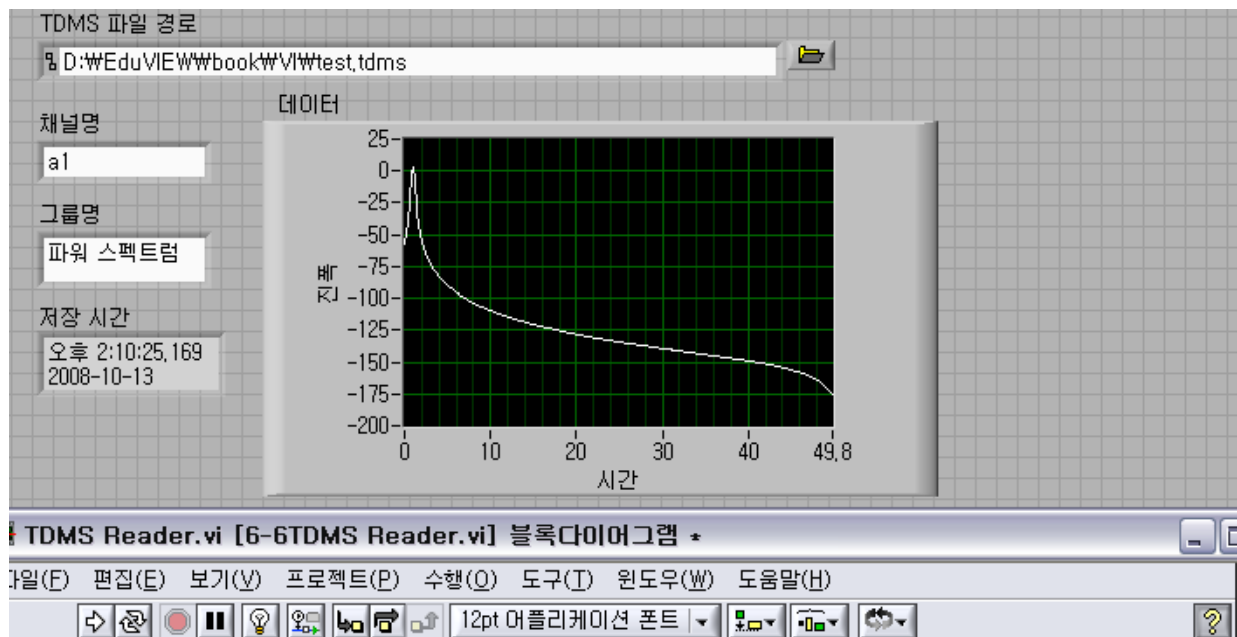




## 실습6-6-1) TDMS 파일 쓰기



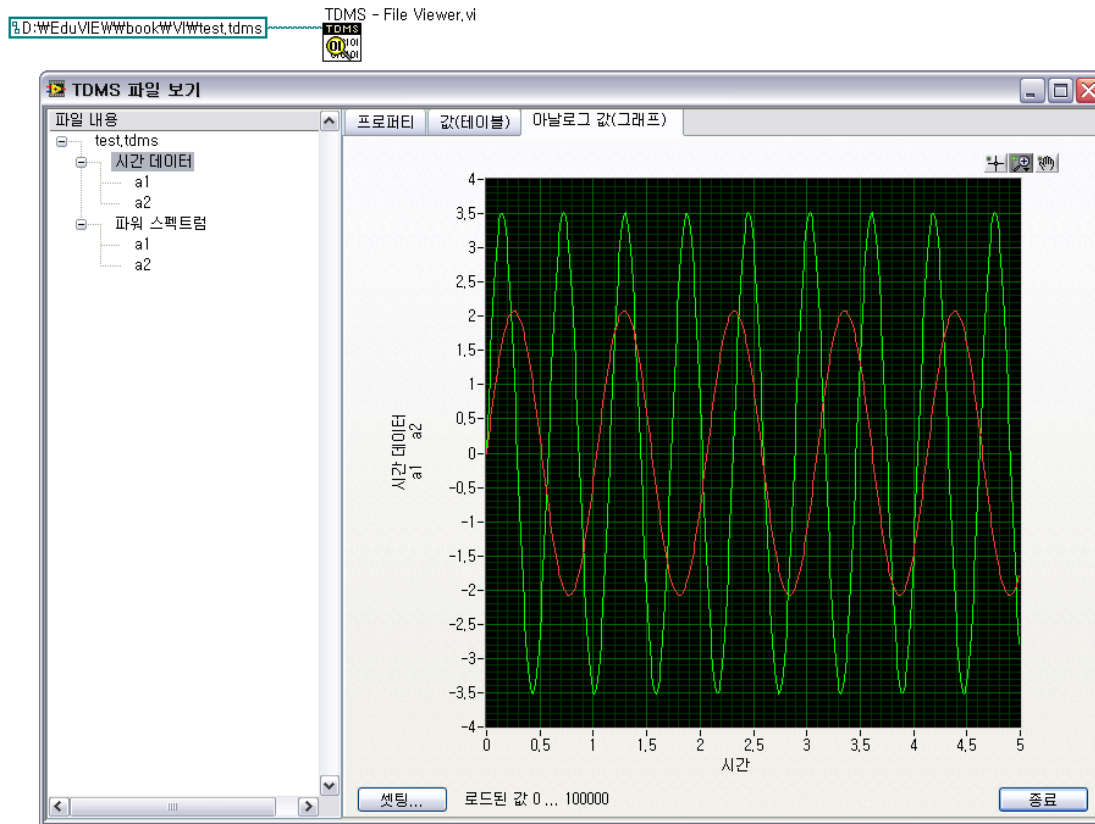
# TDMS 파일 읽기



## **실습6-6-2) TDMS 파일 읽기**



# TDMS 파일 뷰어



- 읽기 프로그램을 따로 작성하지 않아도 'TDMS 파일 뷰어'를 통해 TDMS 파일을 쉽게 읽어올 수 있음.