

Experiment 5:

Introduction to LabVIEW

포항공과대학교 전자전기공학과

Communication and Intelligent Systems Lab. (CISL)

Outline

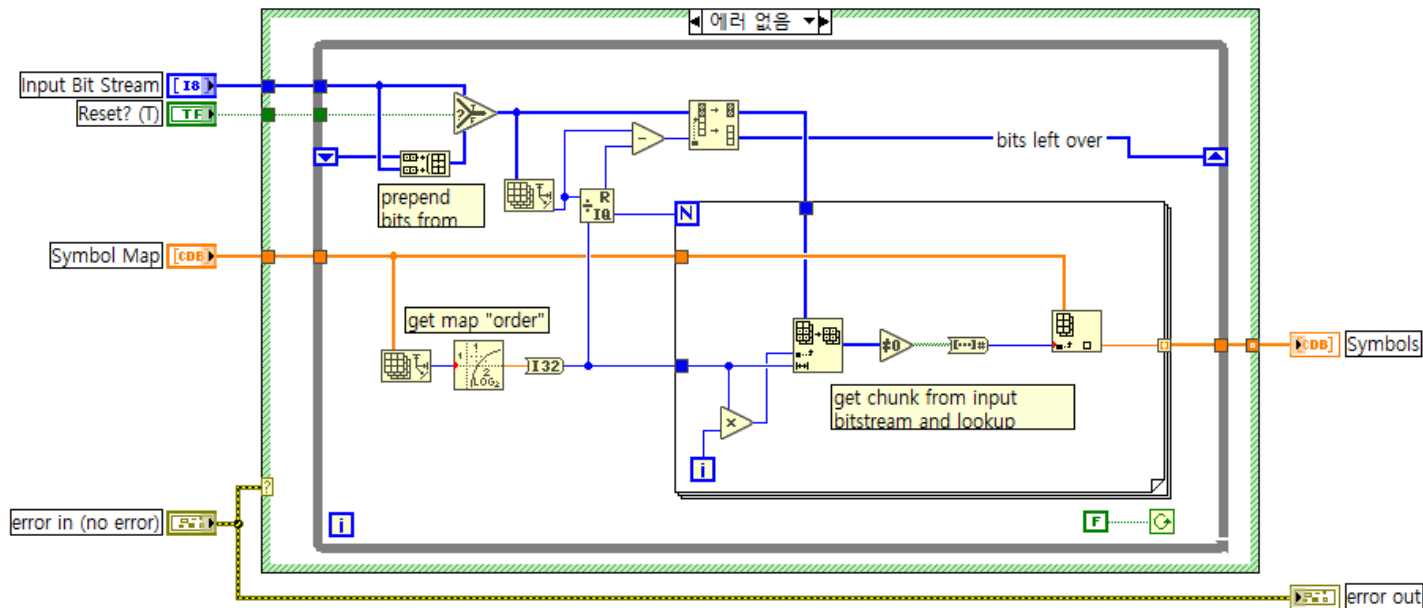
1. Introduction to LabVIEW
2. USRP with LabVIEW
3. Today's Experiment
4. About Report

Outline

1. Introduction to LabVIEW
2. USRP with LabVIEW
3. Today's Experiment
4. About Report

Introduction to LabVIEW

- LabVIEW란?
 - 그래픽 기반의 프로그래밍 언어
 - Hardware 제어에 최적화(especially, NI 장비)



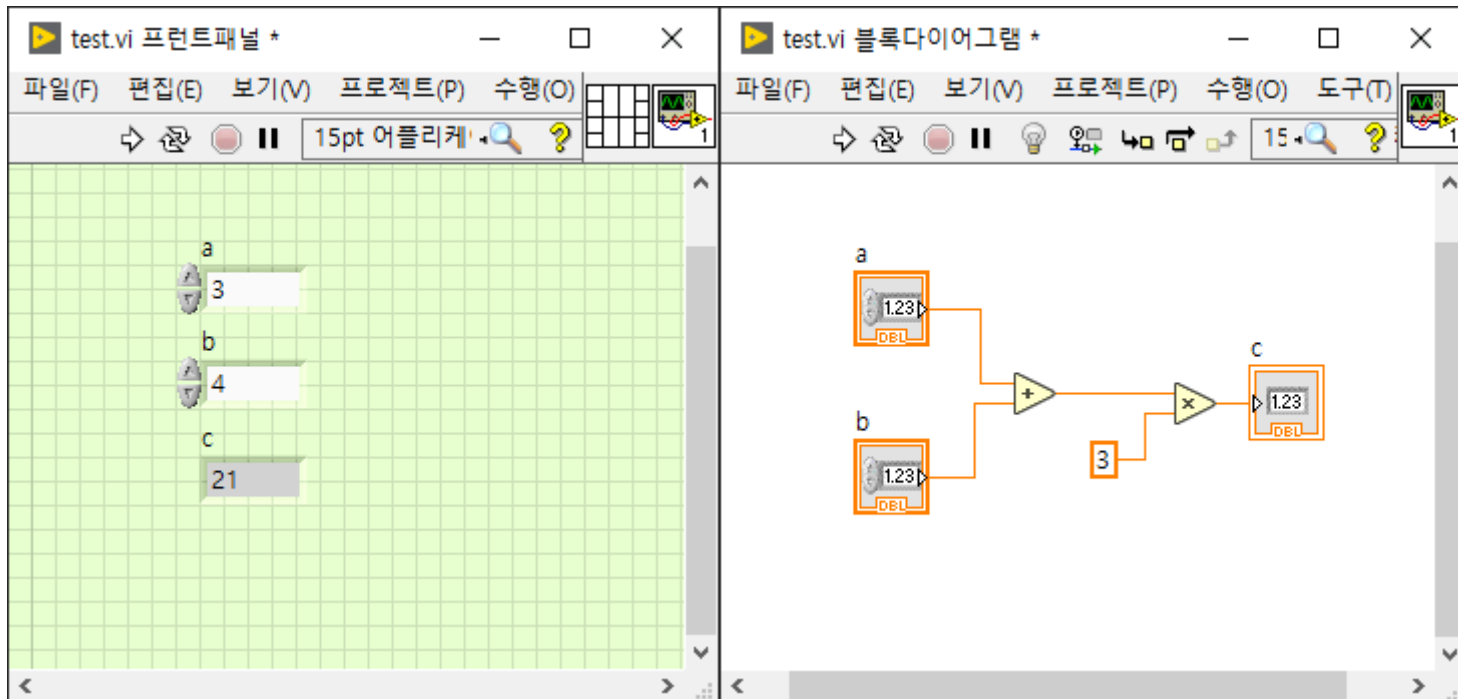
Introduction to LabVIEW

- LabVIEW 시작화면



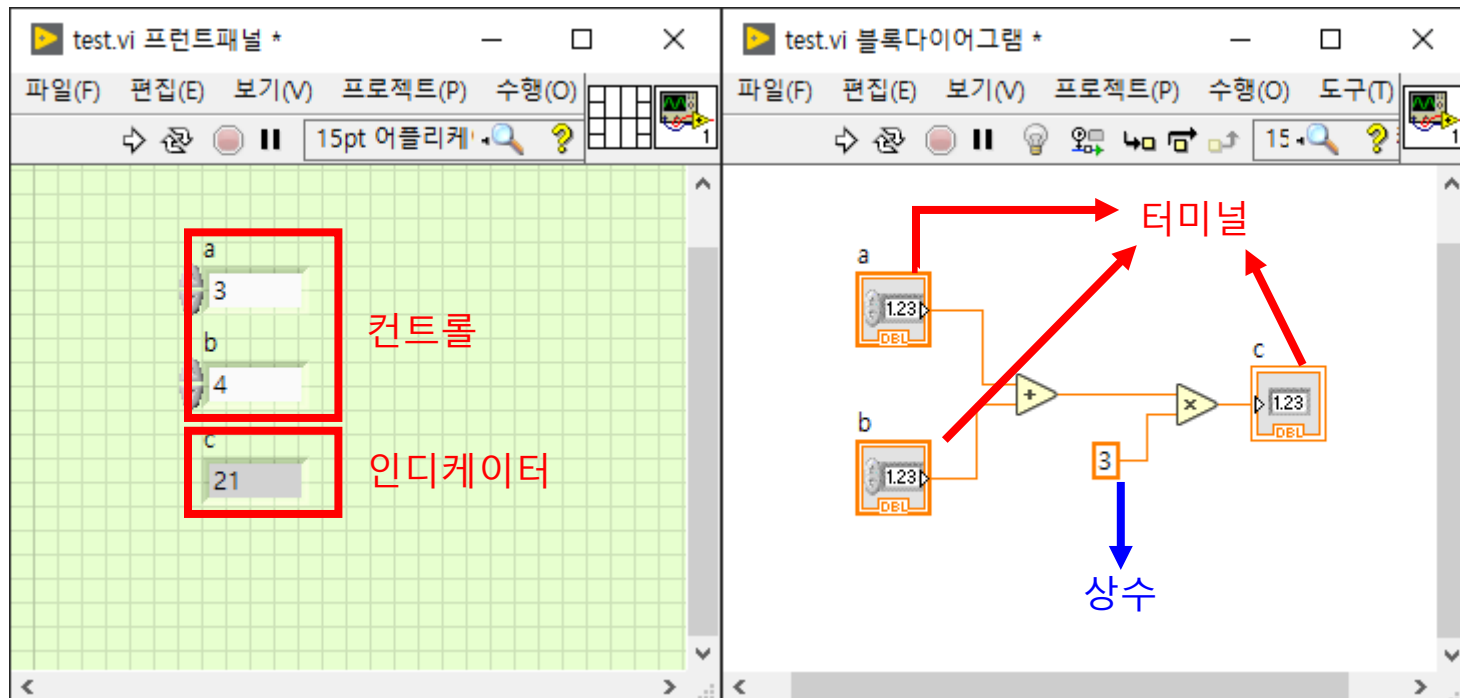
Introduction to LabVIEW

- VI(Virtual Instrument)
 - LabVIEW 프로그램의 최소 단위 (C++/MATLAB의 함수와 유사)
 - 프론트패널과 블록 다이어그램으로 구성



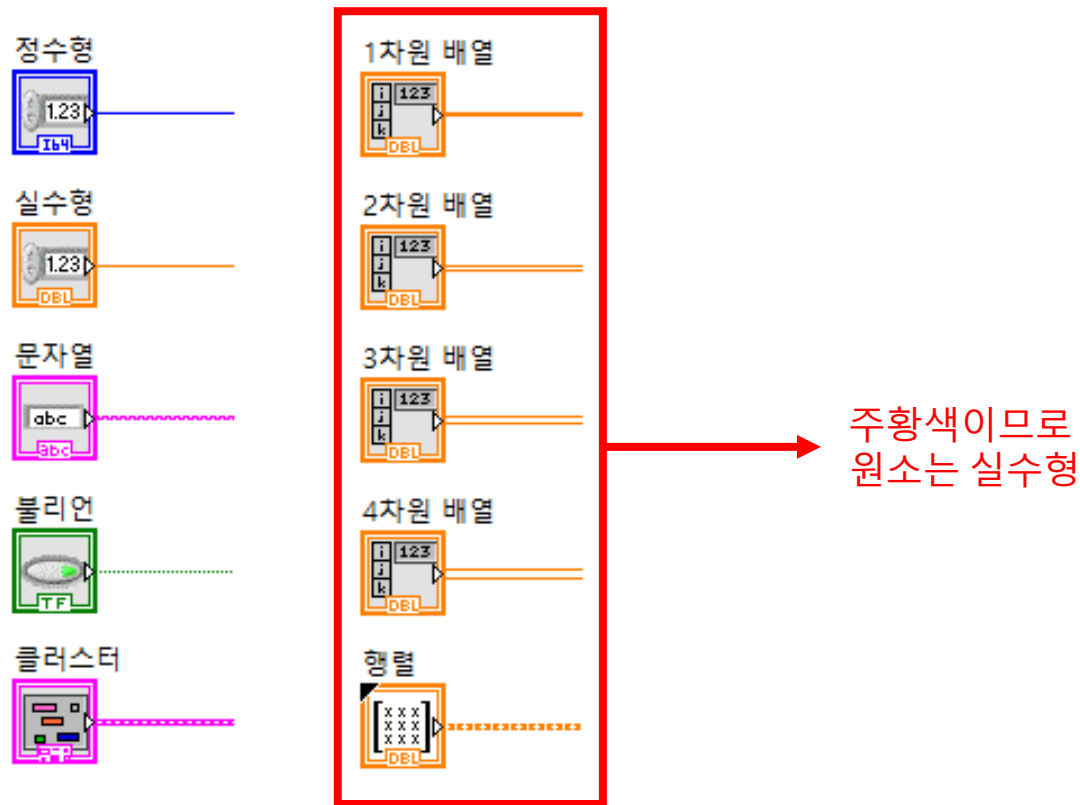
Introduction to LabVIEW

- 변수 : 컨트롤(입력)과 인디케이터(출력)
- 블록 다이어그램에서는 터미널.



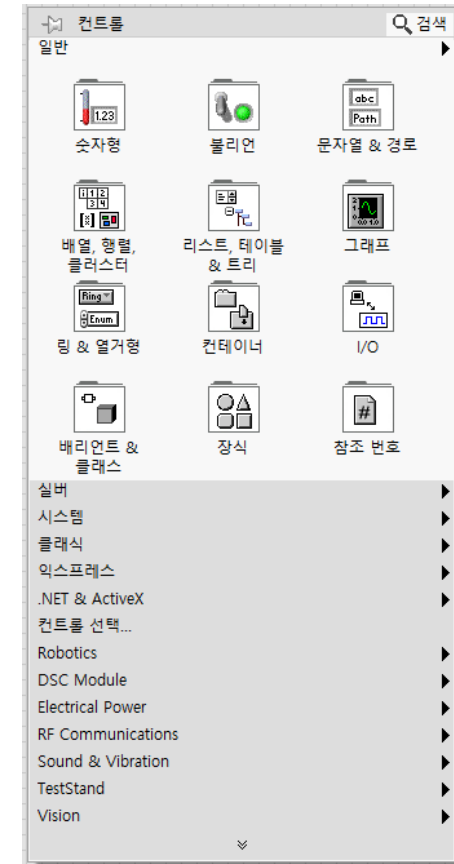
Introduction to LabVIEW

- 변수/상수의 data type은 터미널의 색깔과 와이어로 구분.



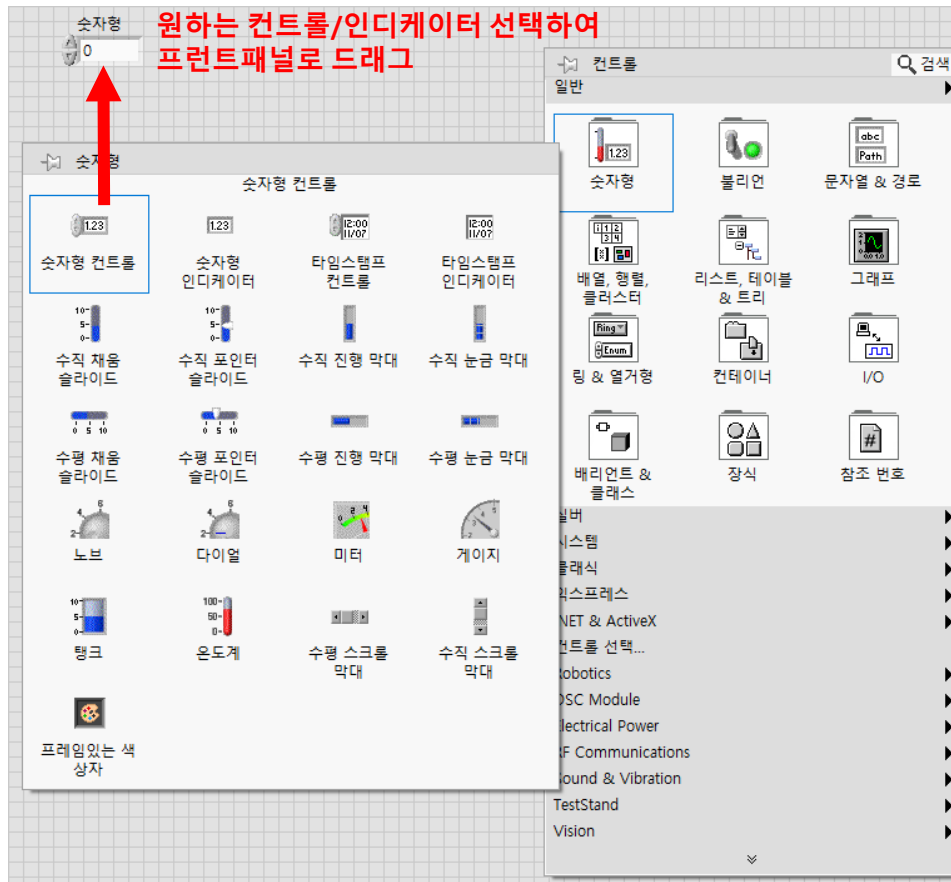
Introduction to LabVIEW

- 컨트롤/인디케이터 찾기
 - 컨트롤 팔레트 : **프런트패널**에서 마우스 우클릭
 - 우측 상단 검색 기능으로 필요한 컨트롤 검색 가능



Introduction to LabVIEW

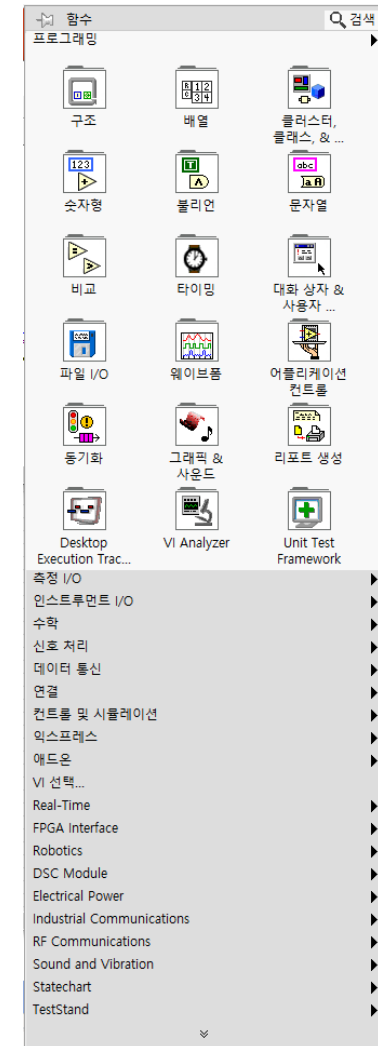
- 컨트롤/인디케이터 배치



대응하는 터미널이
블록다이어그램 상에 자동 배치됨.

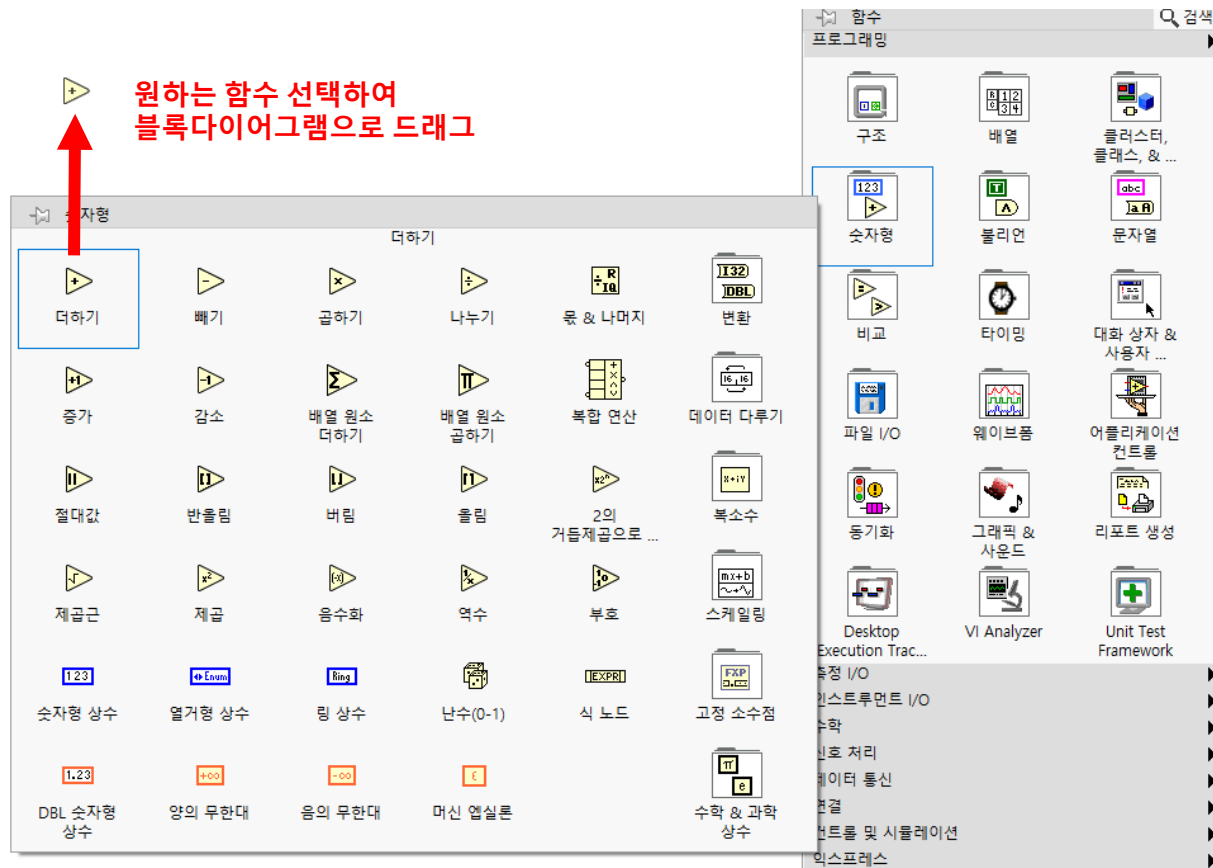
Introduction to LabVIEW

- 함수 : LabVIEW에서 기본으로 제공되거나 사용자가 만들 수 있는 vi
- 함수 찾기
 - 함수 팔레트 : 블록 다이어그램에서 마우스 우클릭
 - LabVIEW에서 기본 제공되는 vi 및 각종 구조들(for, while, case 등) 모음 → 노드
 - 우측 상단 검색 기능으로 필요한 함수 검색 가능



Introduction to LabVIEW

- 함수 배치 및 와이어링

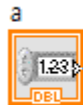


Introduction to LabVIEW

- 함수 배치 및 와이어링



컨트롤/인디케이터/상수의 마우스 포인터가
실패 모양으로 변하면 클릭



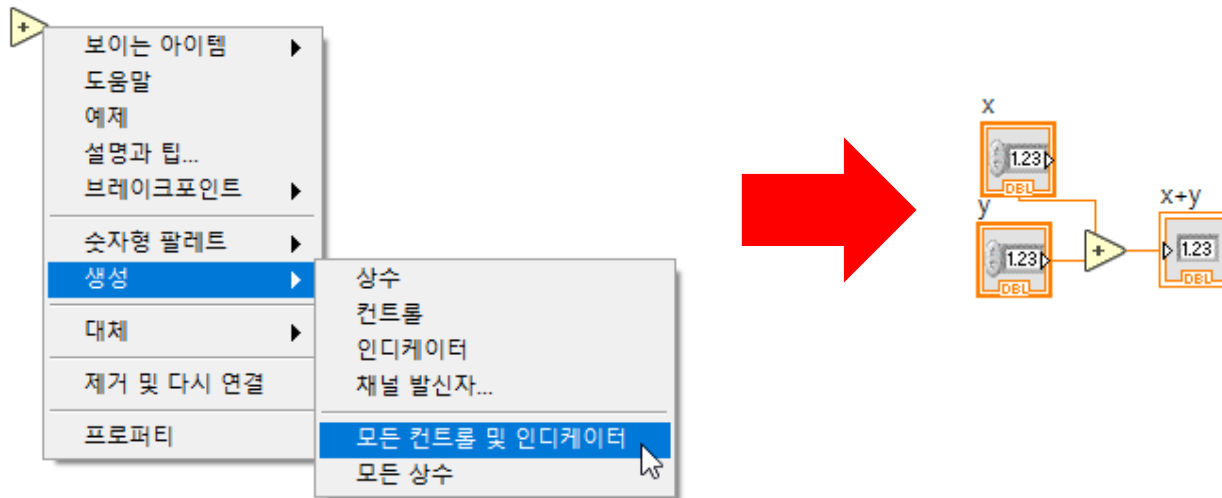
연결하고자 하는 지점으로 마우스 포인터 이동
실패 모양으로 변하면 클릭



와이어링 완료

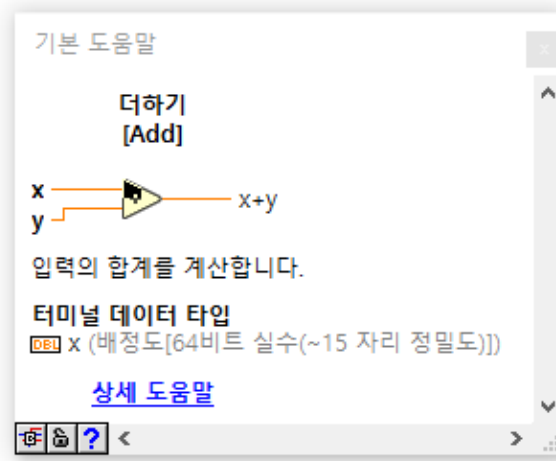
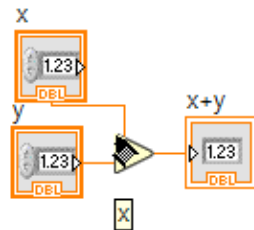
Introduction to LabVIEW

- 함수 배치 및 와이어링
 - 함수에서 마우스 우클릭 → 바로가기 메뉴
 - 함수 동작에 필요한 모든 컨트롤/인디케이터/상수 일괄 생성 가능



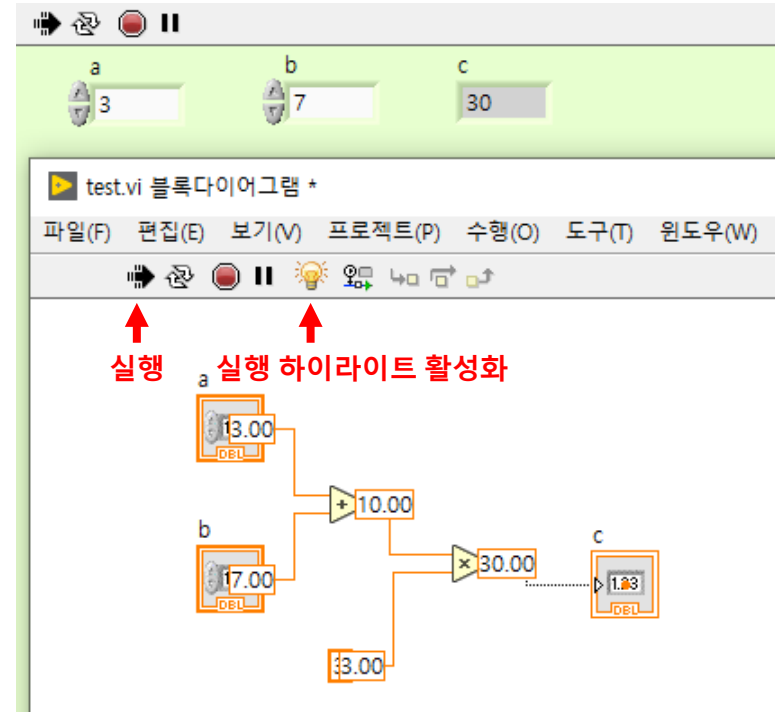
Introduction to LabVIEW

- 기본 도움말(단축키 : Ctrl+H)
 - 함수의 기능을 실시간으로 간략하게 확인하는 데 유용
 - 함수에 마우스 포인터를 위치시키면 해당 함수의 기본 도움말이 자동 출력됨.



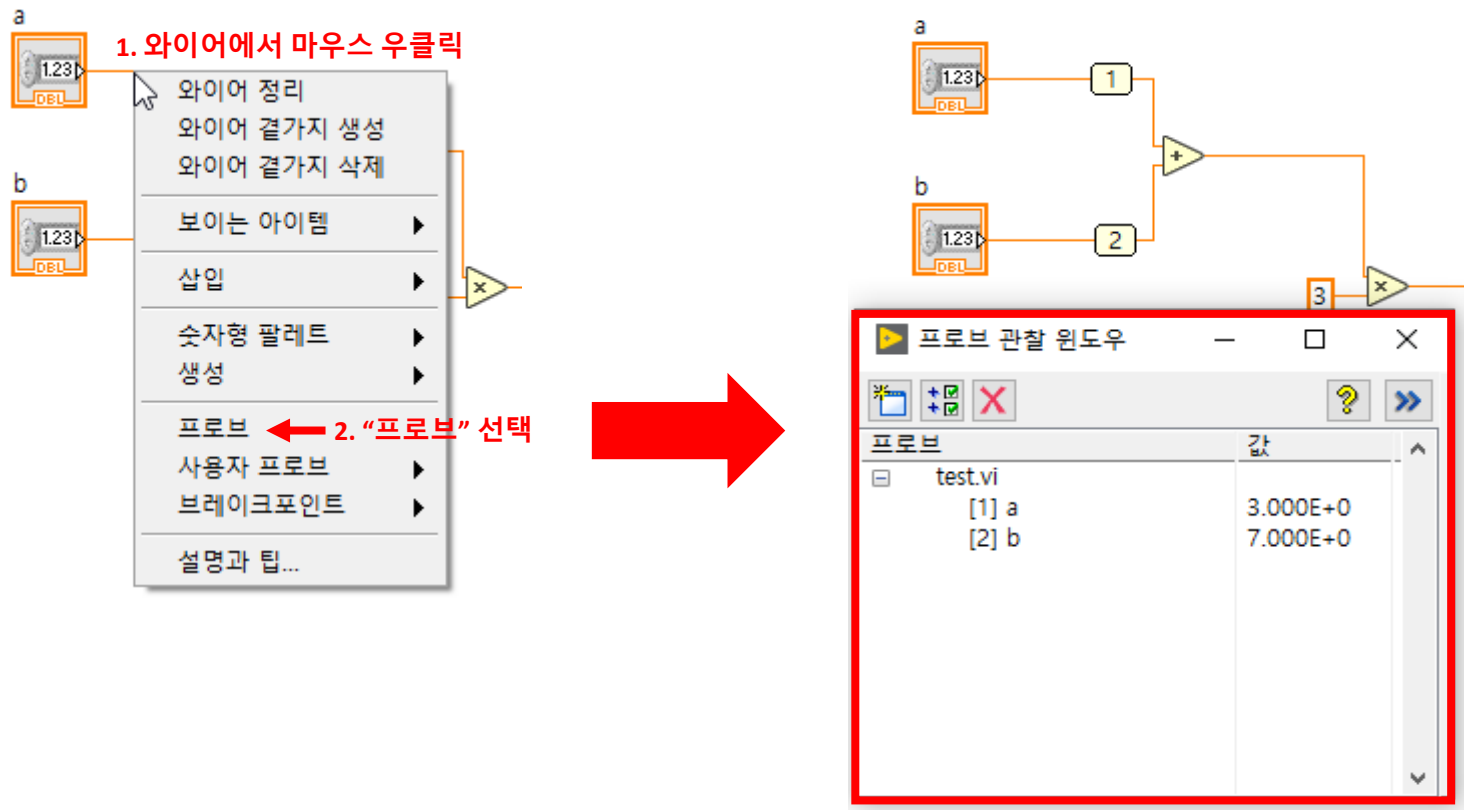
Introduction to LabVIEW

- Vi의 실행
 - 문법적으로 문제가 없으면 프런트패널/블록 다이어그램의 실행 버튼으로 Vi 실행 가능
 - 실행 하이라이트 활성화 시 데이터의 흐름/실행 순서 확인 가능



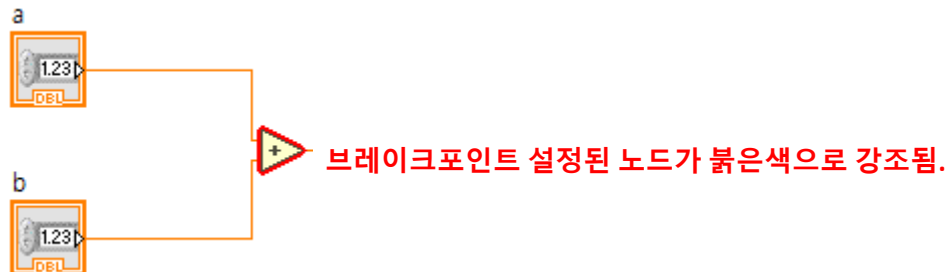
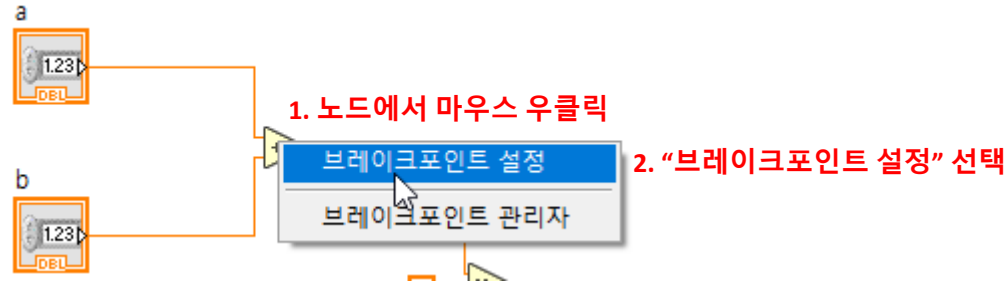
Introduction to LabVIEW

- 프로브 : 와이어 상의 값을 별도의 window에서 모니터링
 - 프로브 관찰 window를 닫으면 **프로브도 전부 제거됨**.



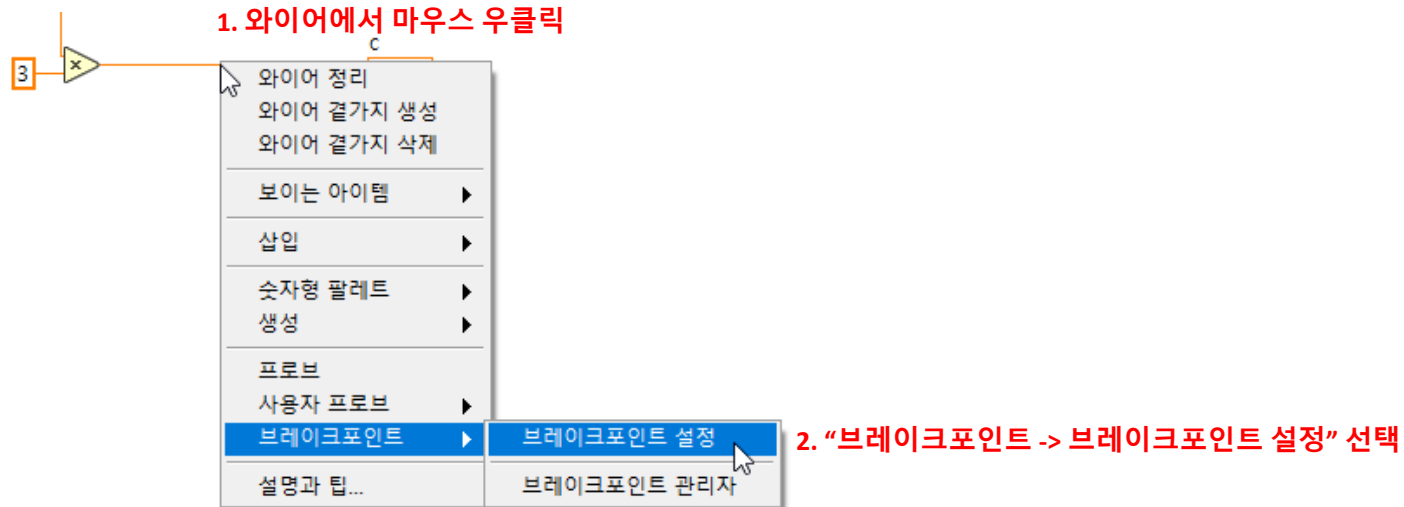
Introduction to LabVIEW

- 브레이크포인트 : 원하는 와이어 혹은 노드에서 프로그램 일시정지
- 노드 브레이크포인트 설정 방법



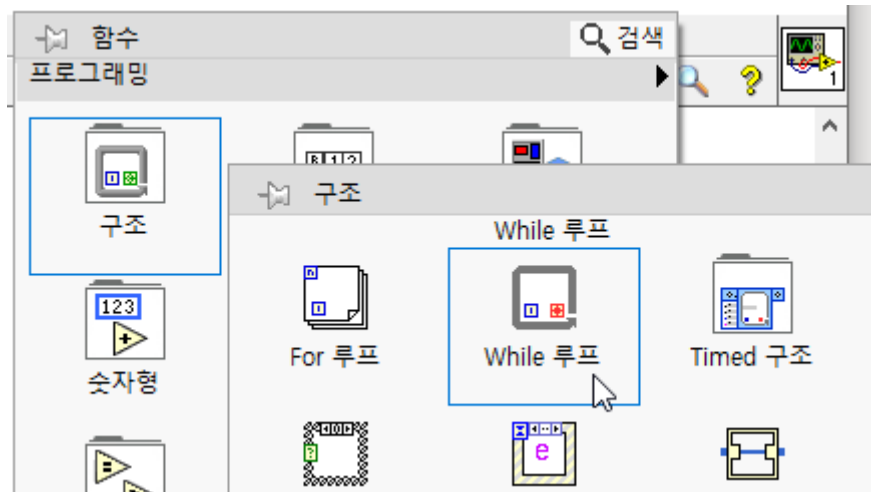
Introduction to LabVIEW

- 와이어 브레이크포인트 설정 방법



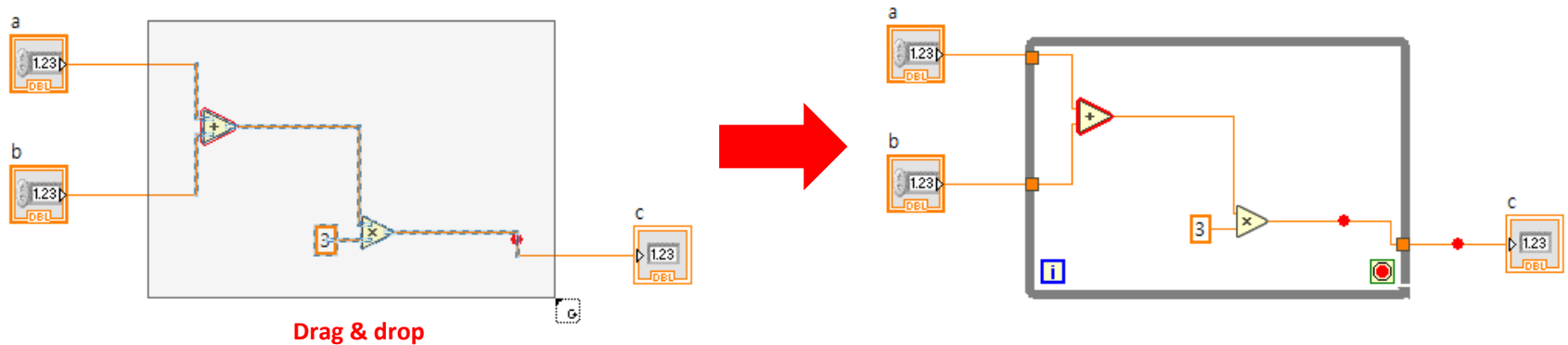
Introduction to LabVIEW

- While 루프 : 루프 내부의 코드를 특정 조건이 만족할 때까지 반복
 - 일단 한 번 실행, 이후 반복 조건 판단. (한 번은 무조건 실행된다.)
 - 함수 팔레트의 “구조” 카테고리 위치.



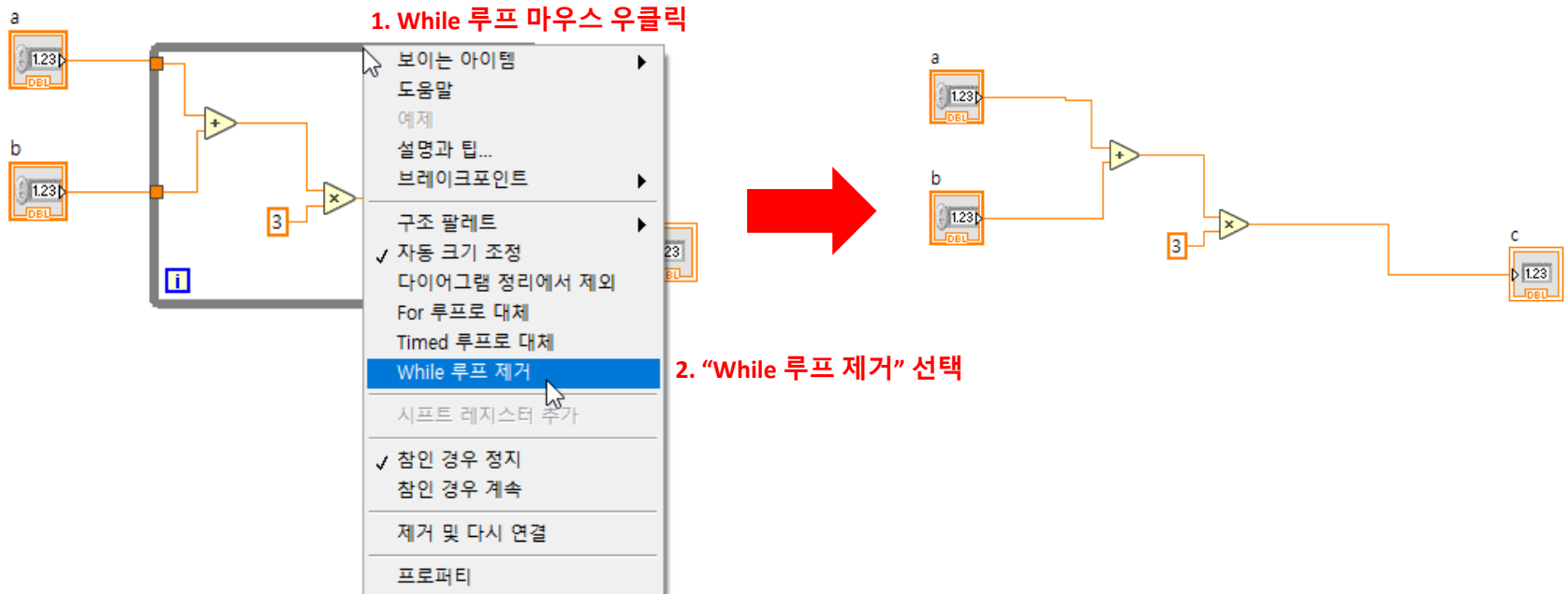
Introduction to LabVIEW

- While 루프 사용법
 - 함수 팔레트에서 “구조 -> While 루프” 선택
 - 반복하고자 하는 구간에 drag & drop



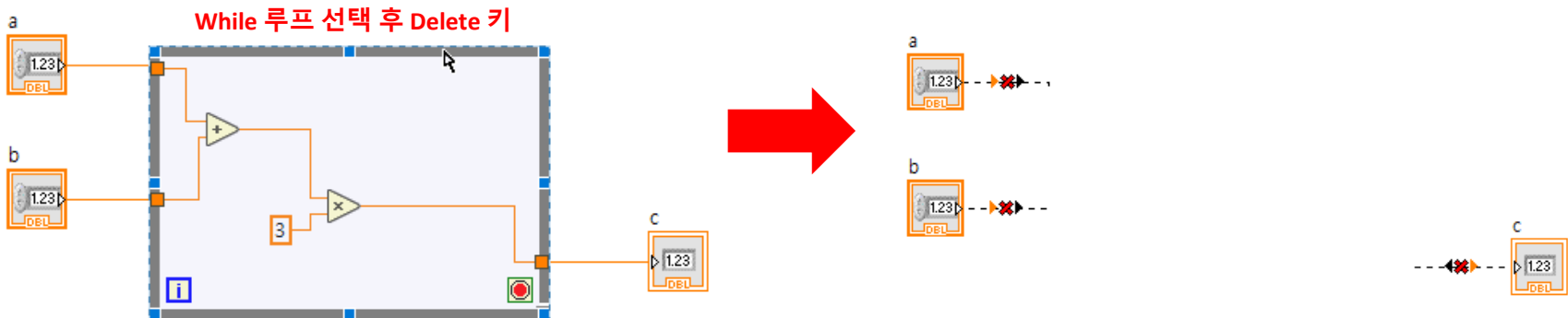
Introduction to LabVIEW

- While 루프 제거 방법
 - While 루프 우클릭 -> “While 루프 제거” 선택



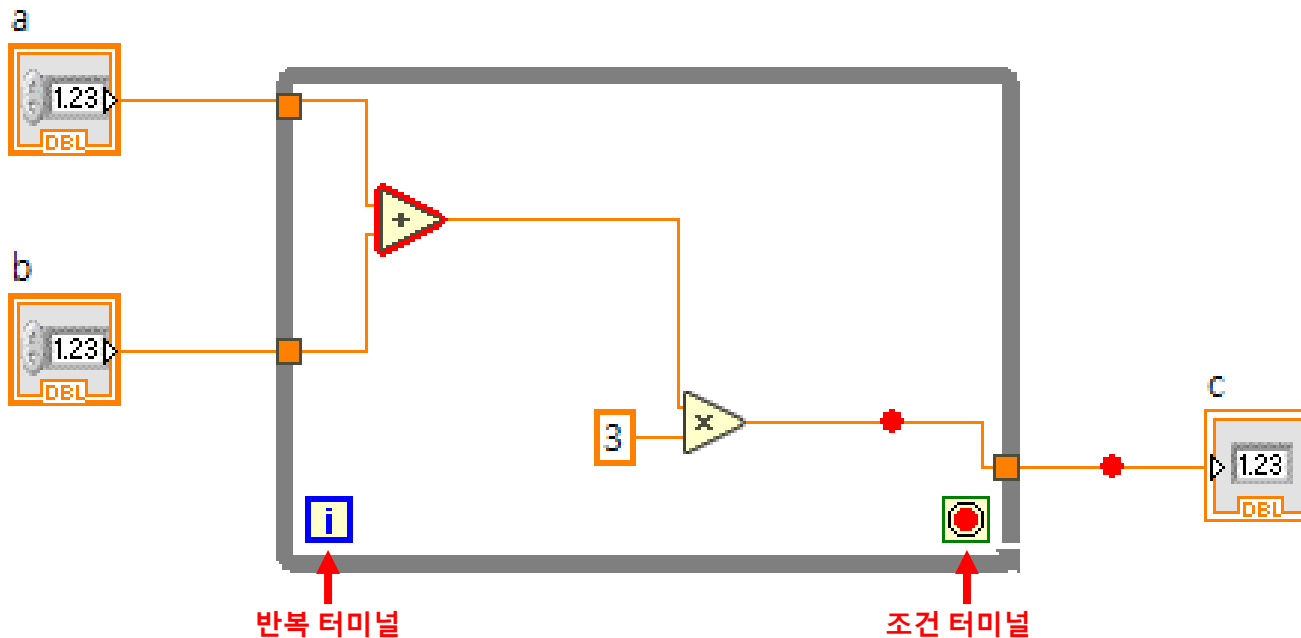
Introduction to LabVIEW

- While 루프 제거 방법
 - While 루프 선택 후 Delete 키를 누르면
While 루프 뿐만 아니라 **내부 구조까지 전부 제거**되니 주의!!



Introduction to LabVIEW

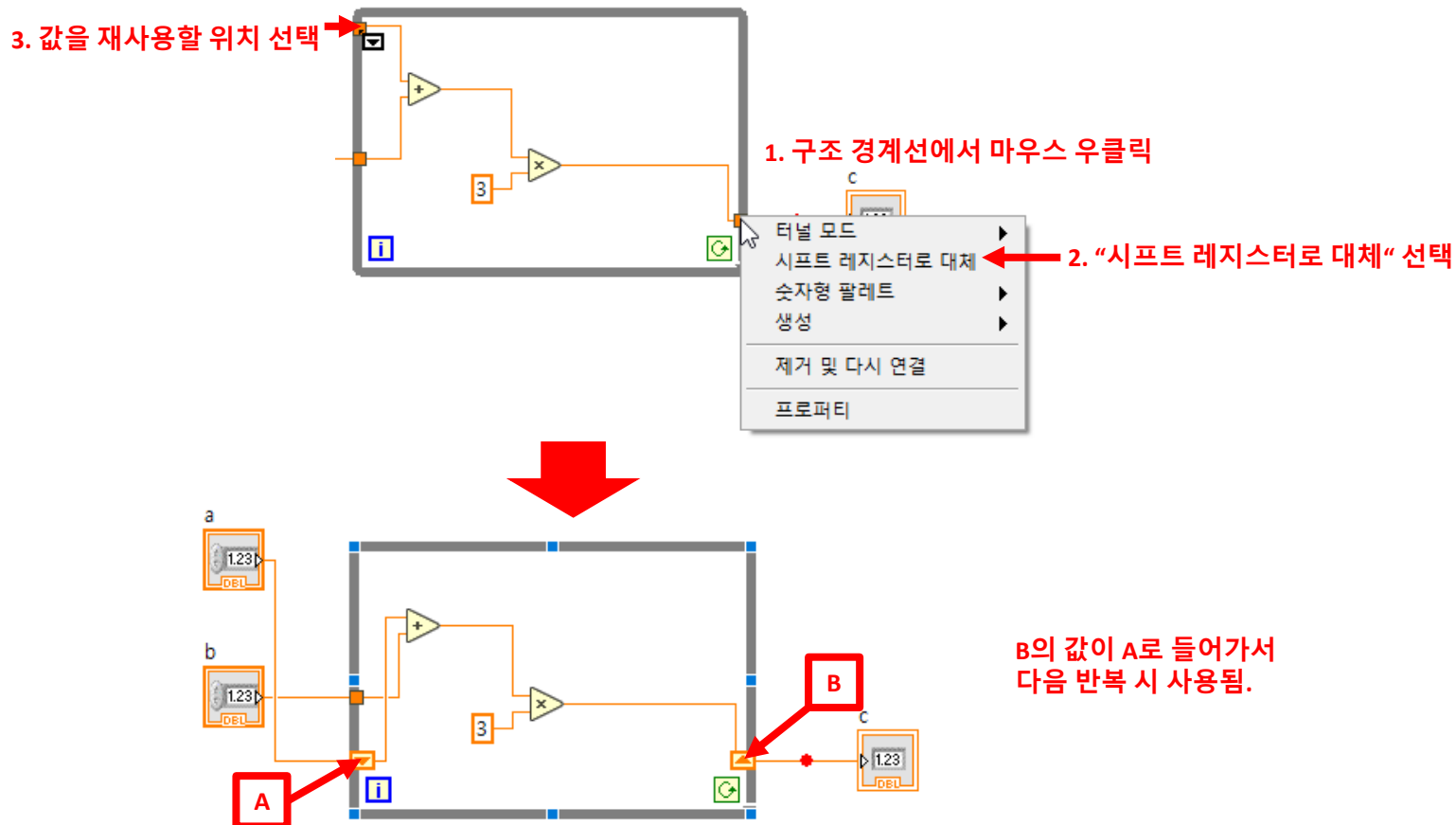
- While 루프 구성요소
 - 반복 터미널 : 반복 횟수 저장
 - 조건 터미널 : 루프 탈출 조건 명시(터미널 클릭하여 변경 가능)
 -  : 조건이 참일 때 탈출
 -  : 조건이 거짓일 때 탈출



Introduction to LabVIEW

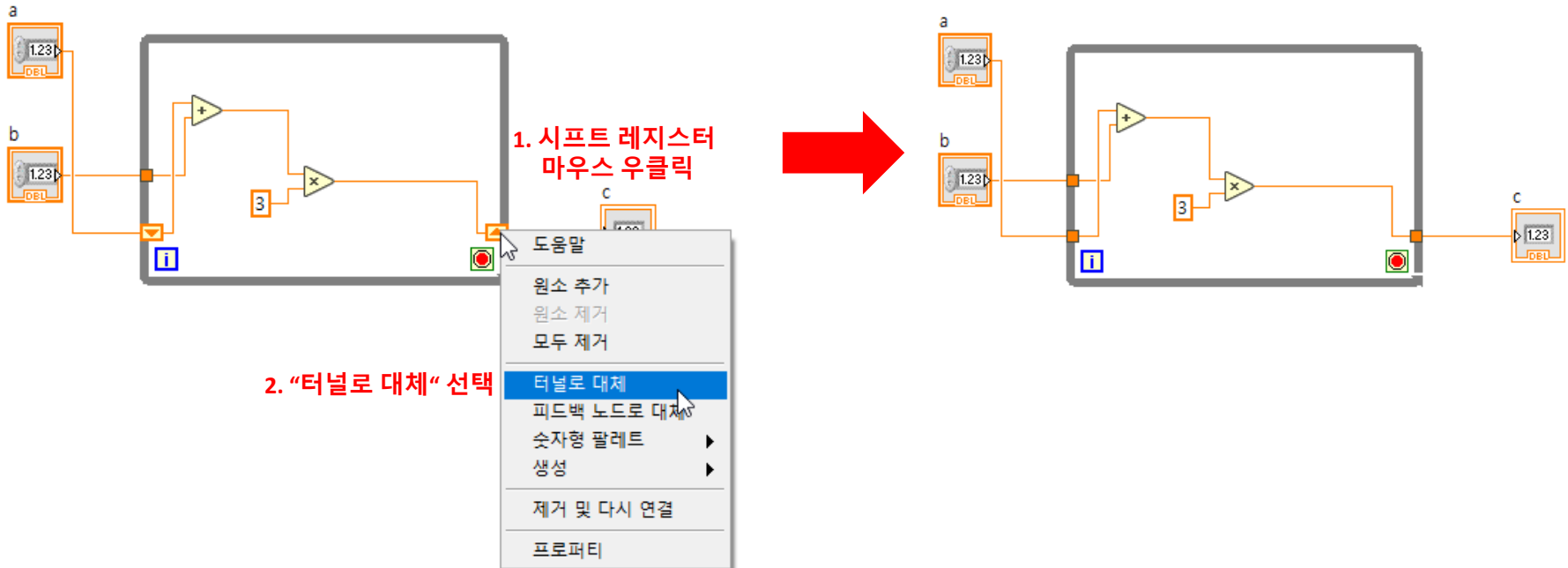
- While 루프 구성요소

- 시프트 레지스터 : 루프 결과값을 다음 반복에서 재사용



Introduction to LabVIEW

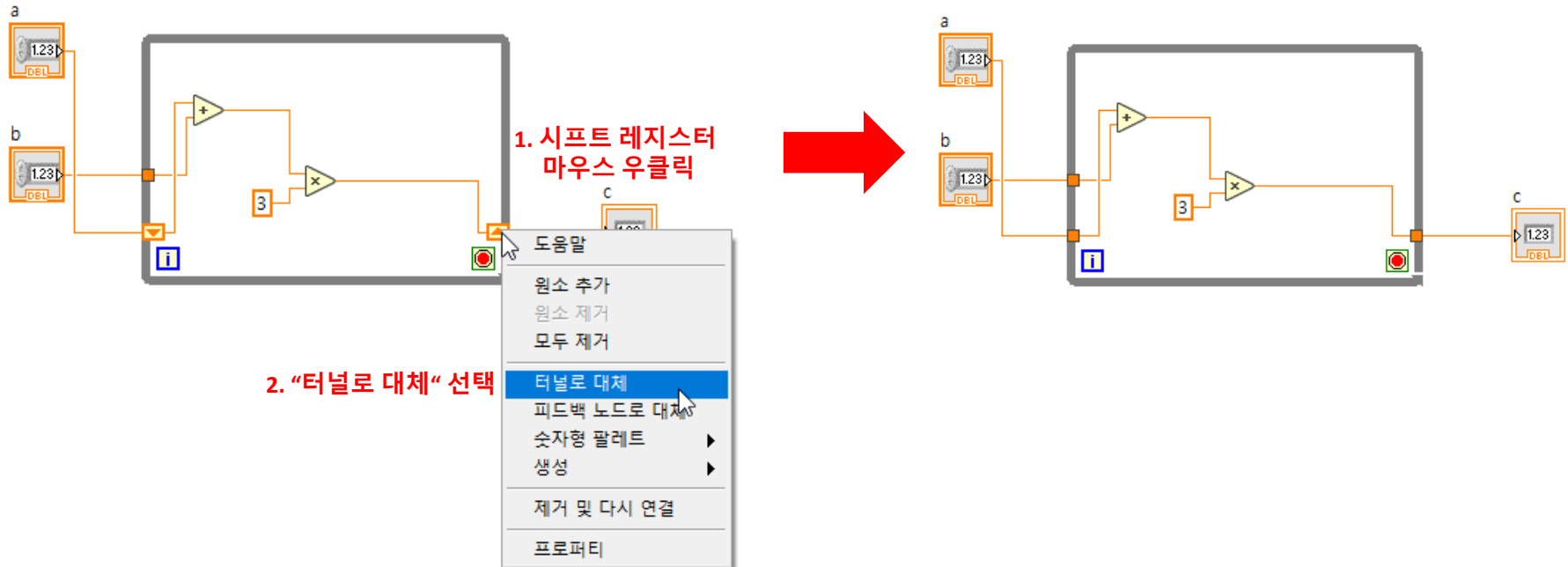
- While 루프 구성요소
 - 시프트 레지스터 제거 방법



Introduction to LabVIEW

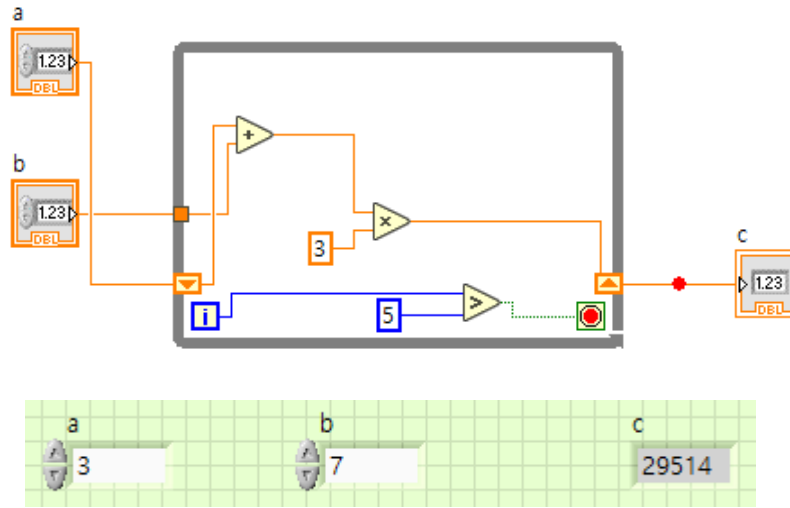
- While 루프 구성요소

- 인덱싱 : 루프의 매 반복마다 터널의 값을 순차적으로 배열로 저장



Introduction to LabVIEW

- While 루프 사용 예시



- 반복 횟수 : 6번

- 반복 횟수(i)가 5보다 클 때 While구조 탈출

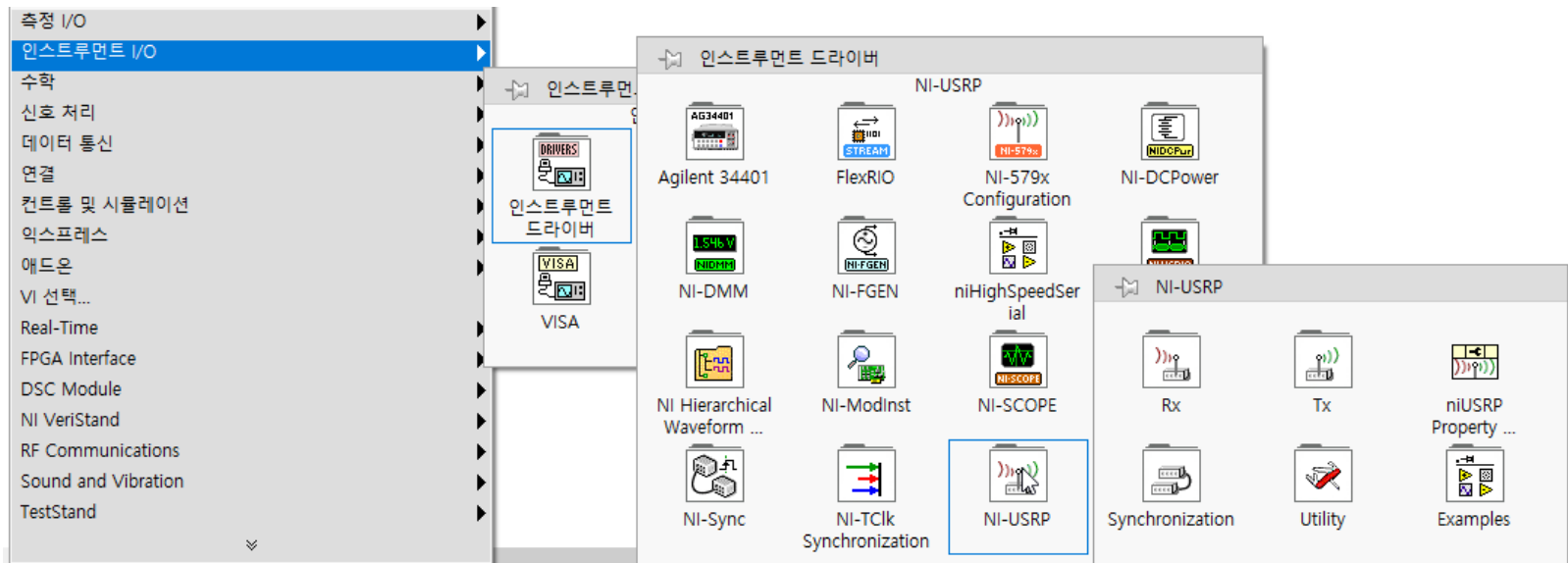
- $(((((3+7)*3+7)*3+7)*3)+7)*3+7)*3+7)*3 = 29514$

Outline

1. Introduction to LabVIEW
2. USRP with LabVIEW
3. Today's Experiment
4. About Report

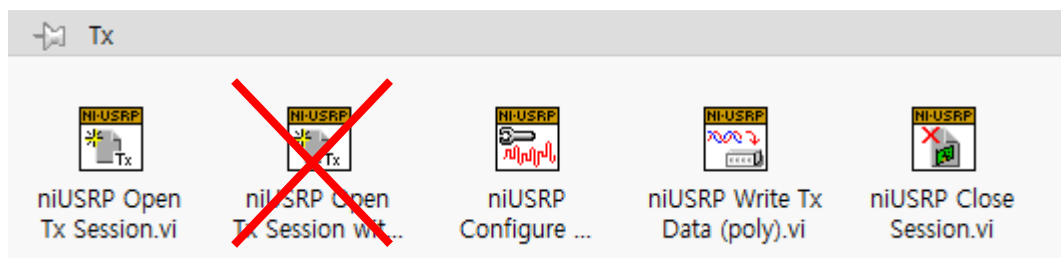
USRP

- LabVIEW의 USRP 관련 function 위치
 - 인스트루먼트 I/O → 인스트루먼트 드라이버 → NI-USRP



USRP

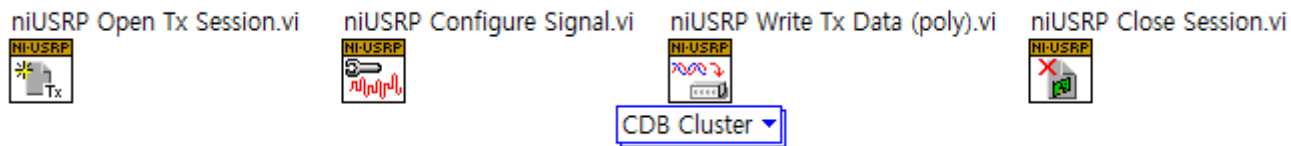
- USRP 송신 관련 주요 function



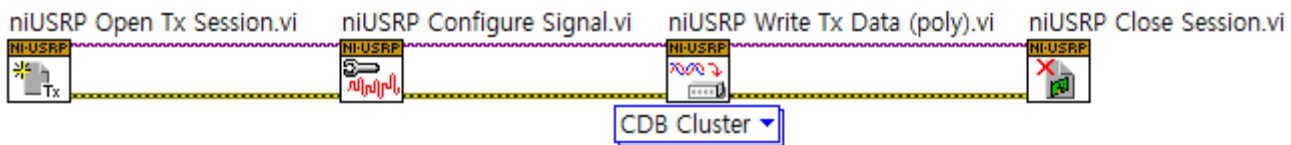
Open Tx Session	USRP로 RF signal을 송신하기 위한 software session을 만든다.
Configure Signal	USRP의 sample rate, carrier frequency, active antenna 등을 설정.
Write Tx Data	전송할 digital sample data를 USRP에 전달.
Close Session	사용중인 USRP session 종료.

USRP

- USRP 송신을 위해서는 4개의 주요 function을 기본적으로 배치해야 함.

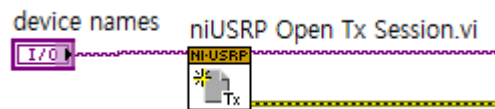


Ctrl 누른 채로 Space bar 한번 클릭 + w 한번 클릭

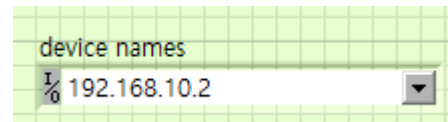


USRP

- 송신에 사용할 장치 지정
 - Open Tx Session.vi에 control 생성하여 Device name 입력
 - 별도 설정 없으면 device name은 기본적으로 “192.168.10.2”로 설정되어 있음.
 - USRP N210의 경우 device name이 “192.168.20.2” 등의 다른 이름을 사용해야 할 수 있으므로 “NI Device Configuration” 프로그램을 통해 확인



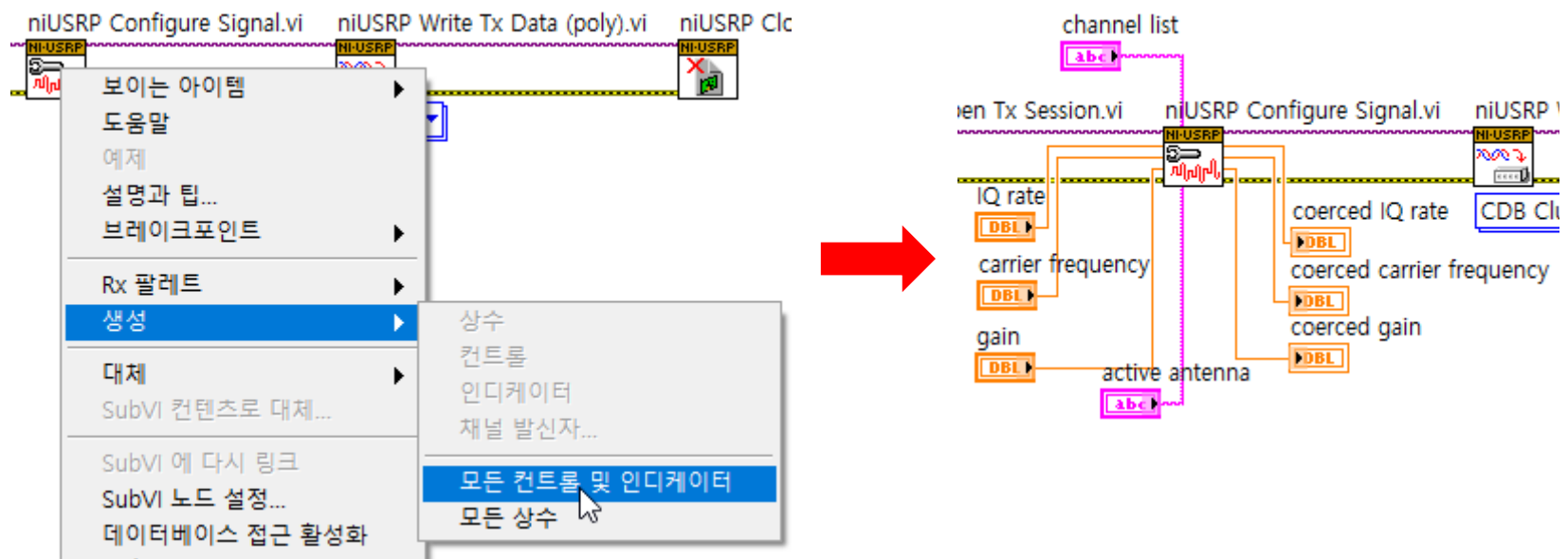
Block diagram에
Control 생성



Front panel에
Name 입력

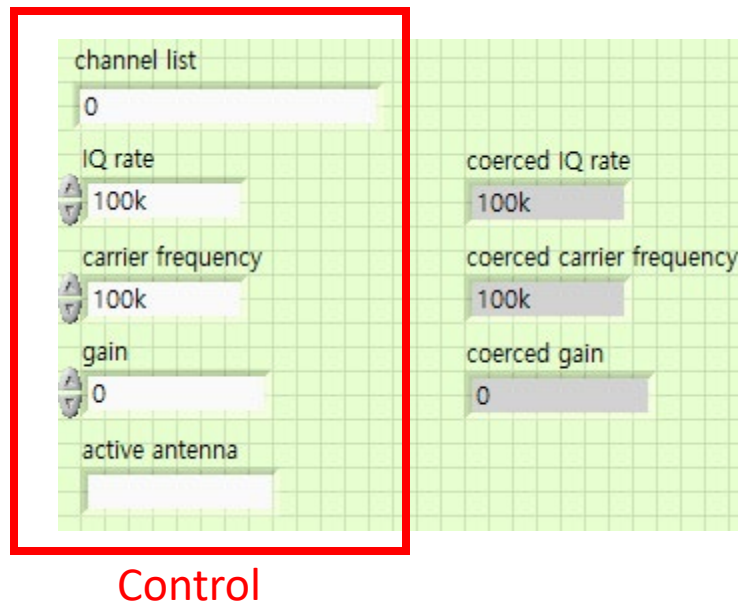
USRP

- USRP 장비의 parameter 설정.
 - Configure Signal.vi 의 모든 control 및 indicator 생성



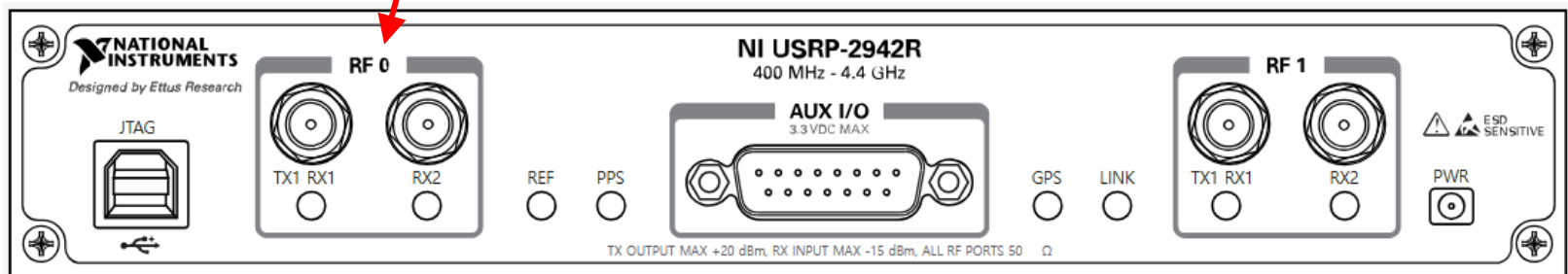
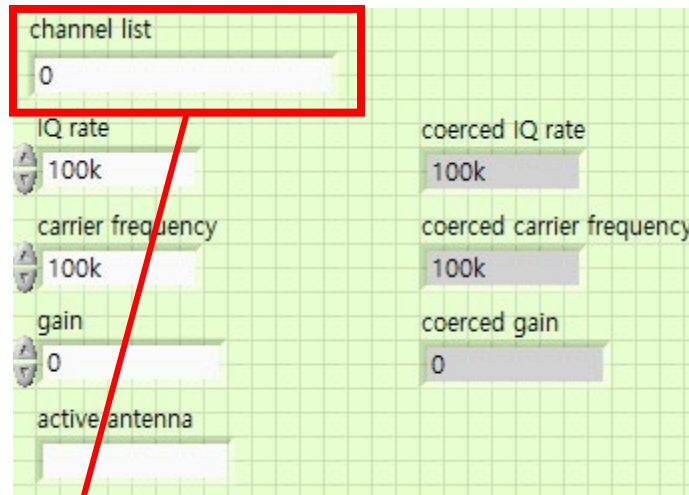
USRP

- Front panel로 이동해서 control에 parameter 값 입력



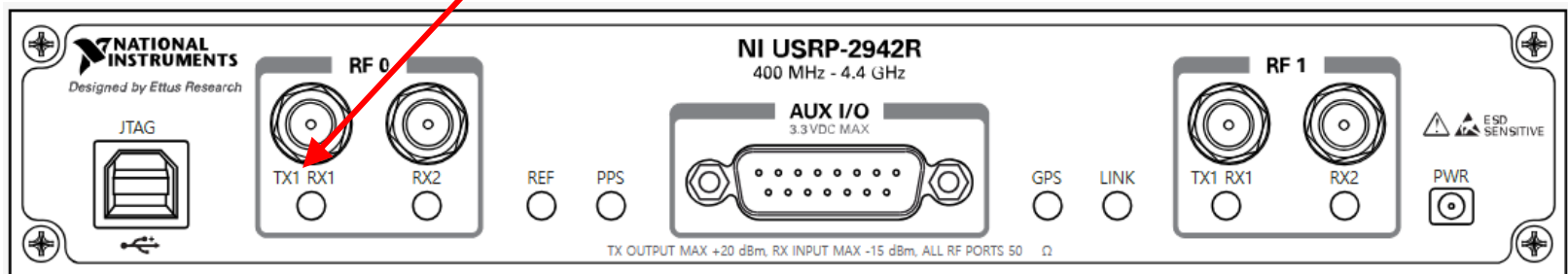
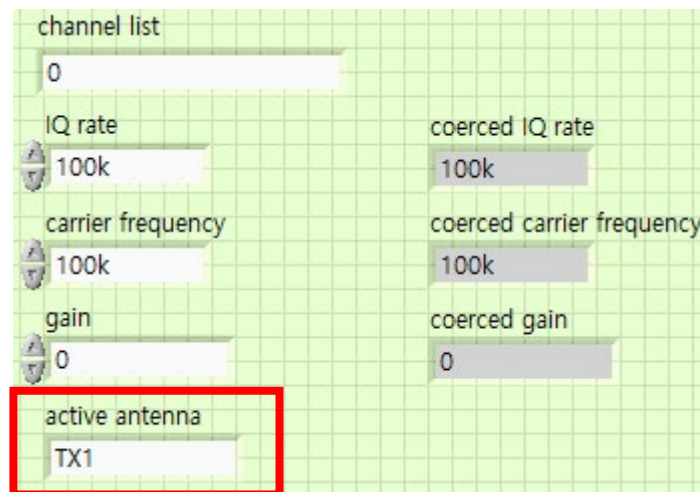
USRP

- Front panel로 이동해서 control에 parameter 값 입력



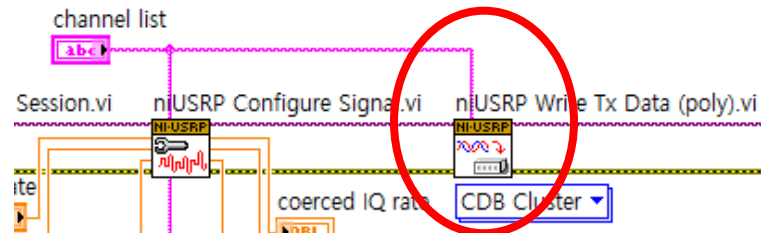
USRP

- Front panel로 이동해서 control에 parameter 값 입력

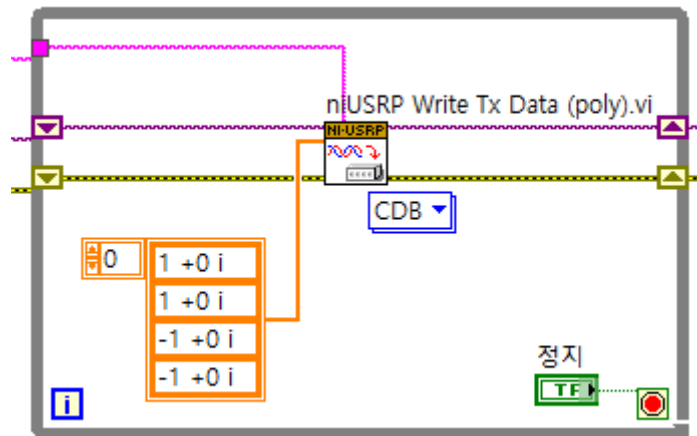


USRP

- Channel list는 write Tx Data.vi에도 입력해줘야 함.

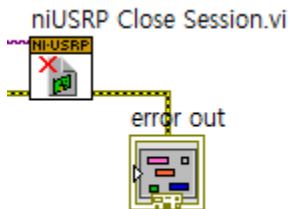


- Write Tx Data.vi에 usrp로 전송할 complex data 입력
 - Real, img value range : -1 ~ 1
 - 입력 후 해당 부분을 loop화하여 지속적으로 전송되도록 함.

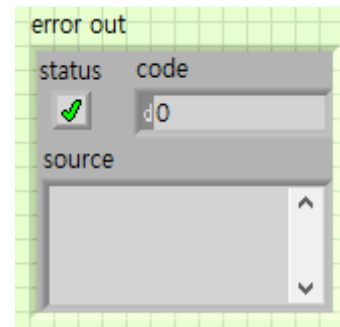


USRP

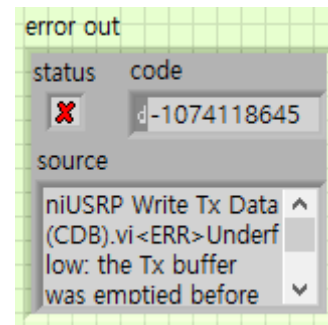
- Close Session.vi에서 에러 cluster를 출력해 프로그램이 제대로 동작하는지 확인
 - Cluster : 서로 다른 data type의 변수를 모아 하나의 변수로!
 - C++의 “구조체”와 유사.



Error 없음
(status : false)



Error 있음
(status : true)

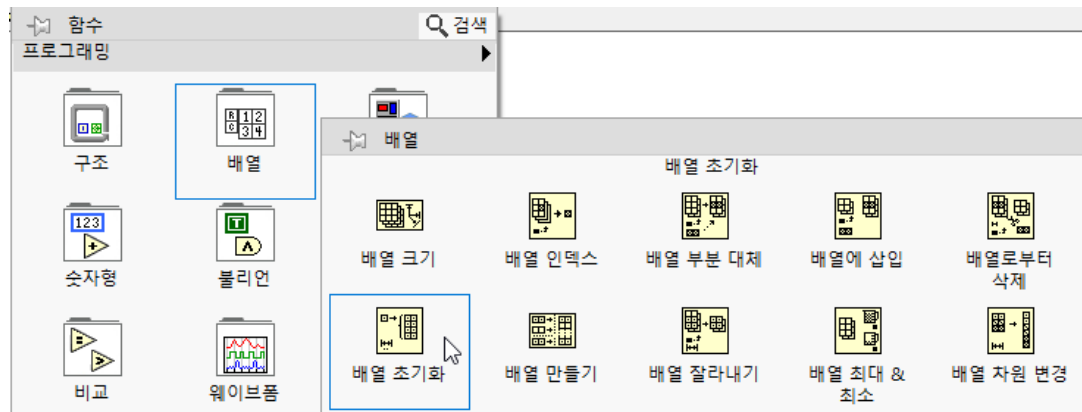


Outline

1. Introduction to LabVIEW
2. USRP with LabVIEW
3. Today's Experiment
4. About Report

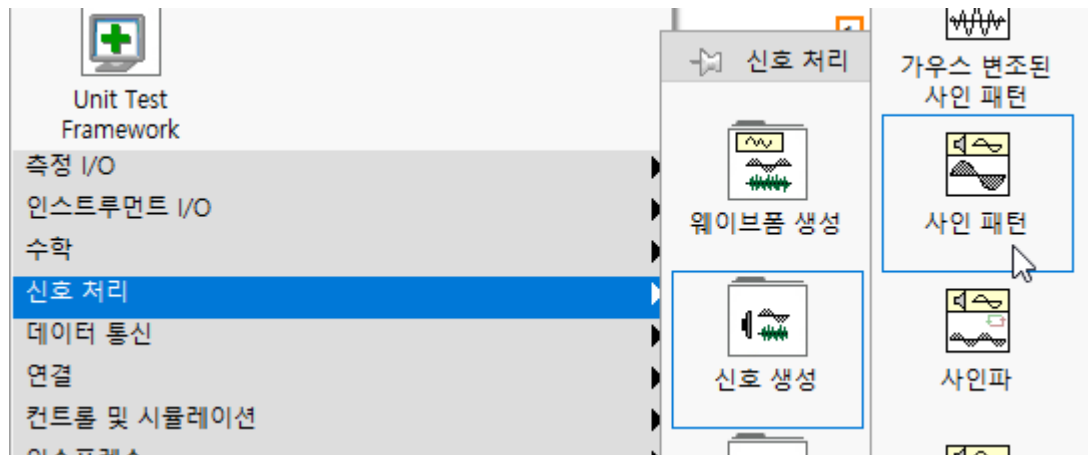
Today's Experiment

- Tone signal 만들어서 전송 ($f_c = 2.3765$ GHz)
 - 배열 초기화 : 특정 원소만으로 이루어진 배열 생성
 - 원소 = 1, 차원 크기 = 100
 - 1이 100개가 들어있는 배열 생성.
 - 생성된 배열을 USRP write TX에 연결
 - 타입 : CDB



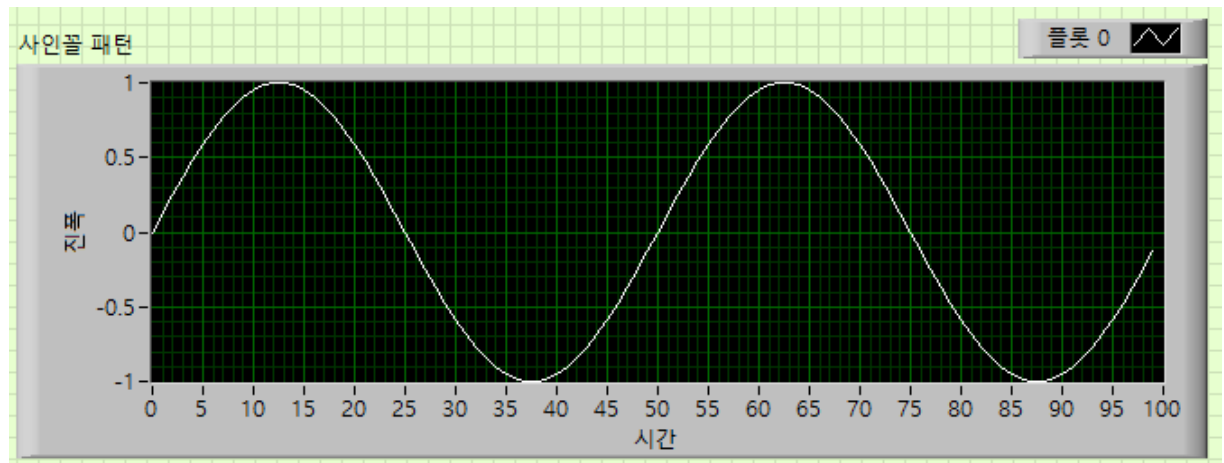
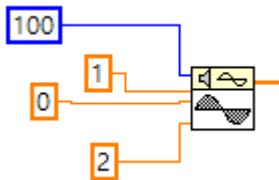
Today's Experiment

- Sine wave 만들어서 전송 ($f_c = 2.3765$ GHz, $10\text{kHz} \leq f_{IF} \leq 20\text{kHz}$)
 - 사인 패턴 : 사인 패턴 배열 생성
 - 생성된 배열을 USRP write TX에 연결
 - 타입 : CDB



Today's Experiment

- Sine wave 만들어서 전송
 - 사인신호 배열 예시
 - 샘플 100, 진폭 1, 위상 0, 사이클 2



Outline

1. Introduction to LabVIEW
2. USRP with LabVIEW
3. Today's Experiment
4. About Report

About report

- Due: 2024. 10. 14 (월) 오후 4:59
- Must be included :
 - Tone signal, sine wave 전송 결과
 - 코드(block diagram), 프런트 패널, spectrum analyzer 결과
 - 설정한 f_c 가 spectrum analyzer에서도 제대로 나오는지 확인
 - 제대로 나오지 않는다면, 그 이유는?
 - $f_c + \Delta f$ 에서 Δf 의 값
 - Spectrum analyzer에서 span을 최대한 좁게 설정.
 - jhchoi0226@postech.ac.kr 로 제출